Technical Report EL-95-28
August 1995

US Army Corps of Engineers
Waterways Experiment Station

Remote Sensing of Surface Unexploded Ordnance at Black Hills Army Depot, Edgemont, South Dakota

by Hollis H. (Jay) Bennett, Jr.

DTIC
ELECTE
JAN 1 9 1996
S D
G

Approved For Public Release; Distribution Is Unlimited

19960105 055

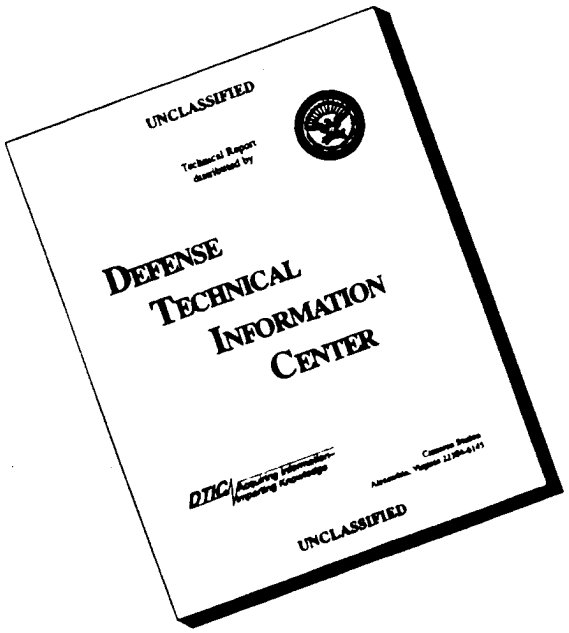
DTIC QUALITY INSPECTED 1

Prepared for U.S. Army Engineer Division, Huntsville

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

PRINTED ON RECYCLED PAPER

DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Technical Report EL-95-28
August 1995

Remote Sensing of Surface Unexploded Ordnance at Black Hills Army Depot, Edgemont, South Dakota

by Hollis H. (Jay) Bennett, Jr.

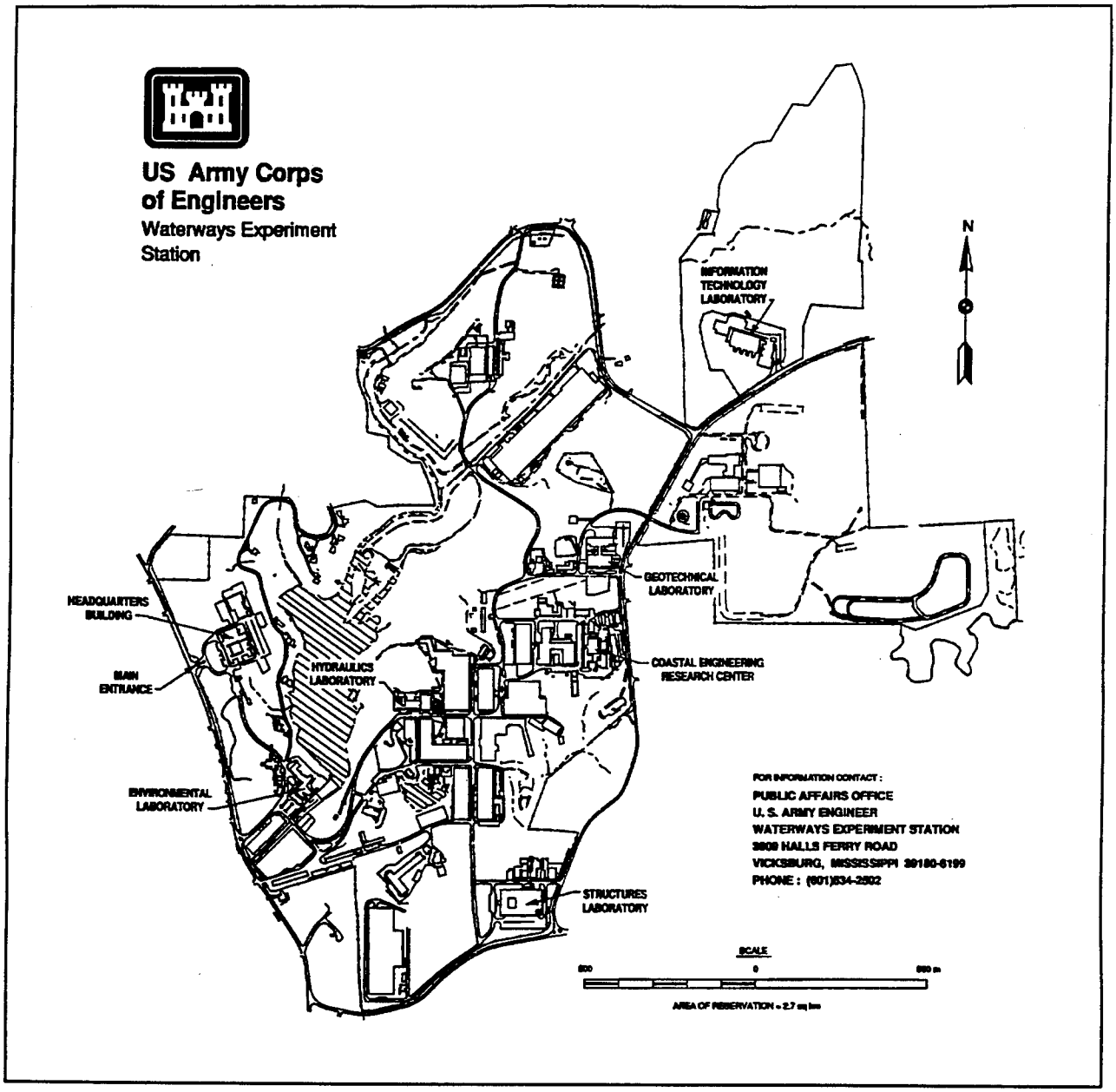
U.S. Army Corps of Engineers
Waterways Experiment Station
3909 Halls Ferry Road
Vicksburg, MS 39180-6199

Accesion For
NTIS CRA&I ☒
DTIC TAB ☐
Unannounced ☐
Justification .....
By .....
Distribution /
Availability Codes
Dist | Avail and / or Special
A-1

Final report

Approved for public release; distribution is unlimited

Prepared for U.S. Army Engineer Division, Huntsville
Huntsville, AL 35807-4301



Waterways Experiment Station Cataloging-in-Publication Data

Bennett, Hollis H.

Remote sensing of surface unexploded ordnance at Black Hills Army Depot, Edgemont, South Dakota / by Hollis H. (Jay) Bennett, Jr. ; prepared for U.S. Army Engineer Division, Huntsville.

156 p. : ill. ; 28 cm. -- (Technical report ; EL-95-28)

Includes bibliographic references.

1. Remote sensing -- Military aspects. 2. Ordnance -- Remote sensing. 3. Remote sensing -- South Dakota -- Edgemont. I. United States. Army. Corps of Engineers. Huntsville Division. II. U.S. Army Engineer Waterways Experiment Station. III. Environmental Laboratory (U.S. Army Engineer Waterways Experiment Station) IV. Title. V. Series: Technical report (U.S. Army Engineer Waterways Experiment Station) ; EL-95-28.

TA7 W34 no.EL-95-28

Contents

Preface v

Conversion Factors, Non-SI to SI Units of Measurement vi

1—Introduction 1

2—Background 2

Black Hills Army Depot Description 2
REMIDS Description 4

3—Site Preparation and Data Collection 6

Base Map and Bench Marks 6
Calibration Site 6
Test Flights 8
Global Positioning System Information 8
Synchronization of GPS and Scanner 13

4—Results 14

Contamination Maps 14
Digital Imagery 17
Archived Flight Data 19

5—Summary 20

References 21

Appendix A: Contamination Maps A1

Appendix B: ARC/INFO GIS Macro Listings B1

Appendix C: Integration of Satellite Information Source Code C1

Appendix D: DAP to GPS Conversion Source Code D1

SF 298

List of Figures

Figure 1. Black Hills Army Depot sites 3

Figure 2. Scanner physical and optical layout 5

Figure 3. Image data of calibration site 7

Figure 4. Test flight paths 9

Figure 5. Test flight coverage map 11

Figure 6. GPS and scanner time synchronization 13

Figure 7. Contamination map coverage 15

Figure 8. Digital imagery 18

Preface

The study herein was sponsored by the U.S. Army Engineer Division, Huntsville, and funded through the Defense Environmental Restoration Program. The study was conducted by personnel of the U.S. Army Engineer Waterways Experiment Station (WES) during July 1993, under the general supervision of Dr. John Harrison, Director, Environmental Laboratory (EL), and Dr. Raymond Montgomery, Chief, Environmental Engineering Division (EED), EL, and under the direct supervision of Dr. Ernesto Cespedes, Acting Chief, Environmental Sensing Branch (ESB), EED.

Mr. Hollis H. (Jay) Bennett, Jr., ESB, prepared this report with significant contributions by Dr. Cespedes. The WES field team included Messrs. Joel Everett, Ricky Goodson, Willie Hughes, and Brian Miles, ESB, and Mr. Henry Blake from the Instrumentation Services Division, WES.

Dr. Robert W. Whalin was Director of WES at the time of publication of this report. COL Bruce K. Howard, EN, was Commander.

This report should be cited as follows:

> Bennett, H. H. (1995). "Remote sensing of surface unexploded ordnance at Black Hills Army Depot, Edgemont, South Dakota," Technical Report EL-95-28, U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.

The contents of this report are not to be used for advertising, publication, or promotional purposes. Citation of trade names does not constitute an official endorsement or approval of the use of such commercial products.

Conversion Factors, Non-SI to SI Units of Measurement

Non-SI units of measurement used in this report can be converted to SI units as follows:

Multiply	By	To Obtain
acres	0.40486	hectares
degrees (angle)	0.01745329	radians
feet	0.3048	meters
inches	2.54	centimeters
knots (international)	0.51444	meters per second
miles (U.S. statute)	1.6093	kilometers

1 Introduction

There is increasing need for dual-use or multiuse technology because of current and anticipated Department of Defense budget reductions. This report describes the use of a helicopter-mounted sensing and processing system, originally designed for remote minefield detection, as a tool for detecting unexploded ordnance at the terrain surface. Detection is based on the remote identification of surface anomalies and materials that may indicate the presence of explosive ordnance contamination. The use of airborne remote detection minimizes the risk to personnel during the environmental assessment and analysis of the site.

The U.S. Army Engineer Division, Huntsville, requested the U.S. Army Engineer Waterways Experiment Station (WES) to use the airborne scanner system to detect surface unexploded ordnance (UXO) at the Black Hills Army Depot. The airborne scanner system used is based on the REmote MInefield Detection System (REMIDS). Huntsville Division personnel selected areas of interest based on suspected contamination at the depot, and the U.S. Army Aviation Technical Test Center (AATTC), Fort Rucker, provided aircraft support. WES personnel operated the airborne scanner and processed the data collected from the test flights to generate contamination maps of the areas of interest.

2 Background

Black Hills Army Depot Description

The Black Hills Army Depot is located near Edgemont, SD. The 21,095-acre[1] facility (TCT-St. Louis 1993) was constructed as a reserve depot in 1942 that provided for the maintenance, storage, renovation, and demilitarization of ordnance, ordnance components, and bulk explosives containing high explosive, incendiary, or chemical fillers. Huntsville Division personnel indicated that primarily burning grounds and storage areas are suspected of being contaminated with ordnance and explosive waste (OEW) or chemical warfare material. These areas are shown in Figure 1 and are described below.

There are three areas that are burning grounds. Burning Ground 1 consists of one 438-acre site and several smaller areas totaling about 495 acres, with known UXO present. Prior to 1946, Burning Ground 1 was used for the destruction of white phosphorous, mustard, and conventional ordnance; numbers of disposed ordnance are not available. Burning Ground 2 (Area 5000) was constructed in 1946 as a facility for heavy demolition and destruction of toxic gas-filled ammunition and is approximately 965 acres. Burning Ground 3 (Area X) consisted of approximately 675 acres. The burning ground was used for burning of small arms, conventional ammunition, fragmentation bombs, ammunition components, propellants, bulk explosives, and guided missile fuels and oxidizers.

Three igloo blocks are of interest. Igloo Block G is known to have been the main storage area for V and G nerve agents. Igloo Block F had an explosion on 31 March 1950 that blew off the top of Igloo F-1304, shattering one wall and scattering grenades throughout the area. Because of the presence of hand and rifle grenades scattered throughout the site, the area was fenced and barricaded. Quantity distance tests were conducted in Igloo Block D in 1957, at which time approximately 4,000 M61 rockets containing nerve agent stimulant (ethylene glycol) were placed in igloos and detonated.

1 A table of factors for converting non-SI units of measurement to SI units is presented on page vi.

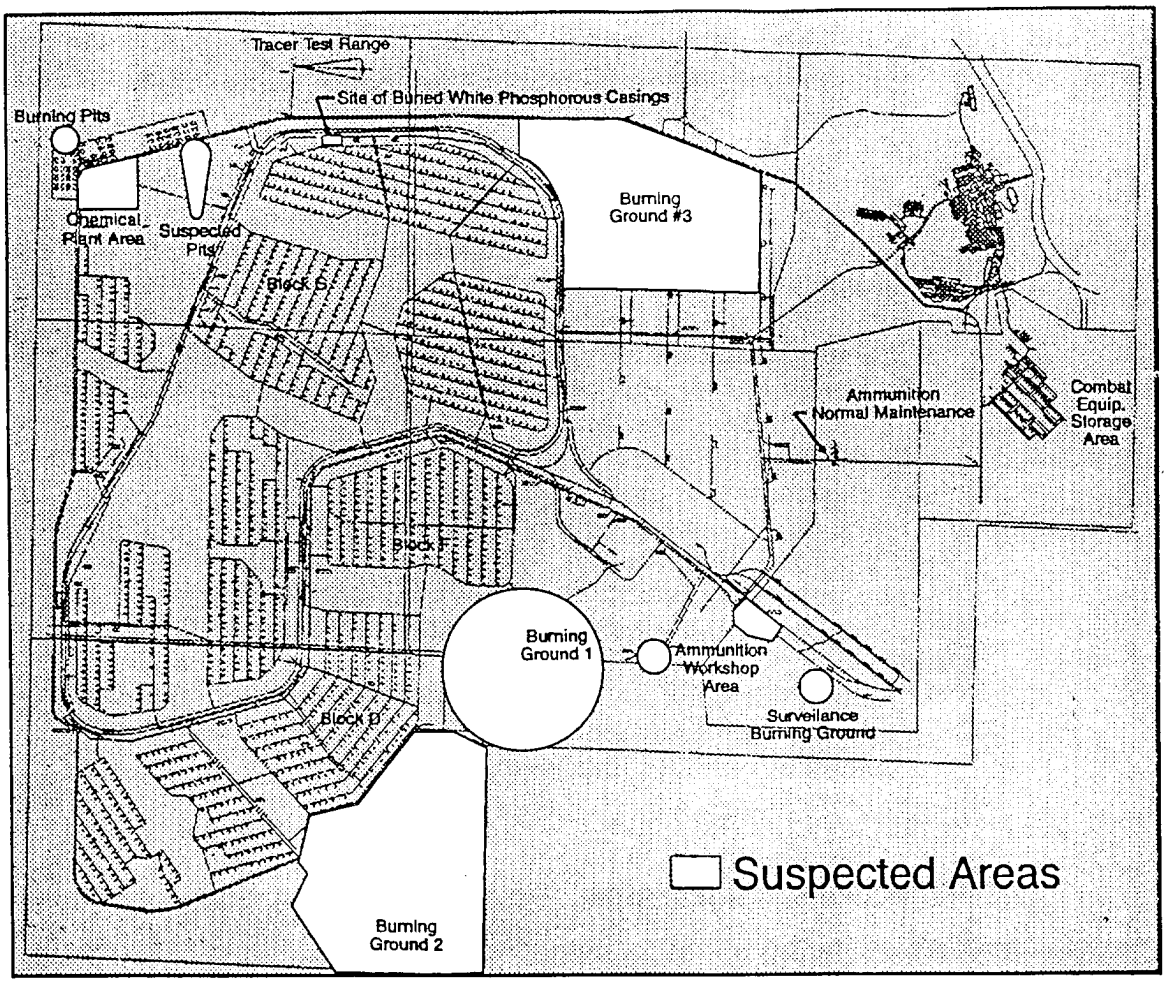


Figure 1. Black Hills Army Depot sites

Eight other target areas are also of high interest:

a. Chemical Area (Area 6000) covers 114 acres. It consists of two separately fenced areas that included a chemical plant, storage area, a chemical burning pit, and outdoor storage facility. According to documentation, the 6000 Area was used for the disposal of mustard, cyanogen chloride, and phosgene bombs.

b. Burial Site (North of Igloo Block J) is approximately 2.8 acres. White phosphorous casings were buried in two pits located north of Igloo Block J.

c. Tracer Test Range (Area 9000) has little information available about it. The area was, however, shown as a restricted area on old maps, and the Statement of Clearance described it as a "nonuse restricted area."

d. Surveillance Area is approximately 4.4 acres. Spent igniter tubes and primers have been found in the area, and a partially denuded portion suggests some type of disposal activity took place.

e. Ammunition Work Shop Area (Area 3000) is a 48-acre site that is currently used for cattle grazing. Of primary concern are the washout facility and leaching beds. UXO have been found in the area. The area was used for extracting explosives and renovation of ordnance.

f. Combat Materials Area (Area 1800 and 2000) was used to store and reassemble small arms munitions. Additionally, inert materials and salvage items were stored at this location. A salvage yard landfill and burning pit for inert and nonsalvageable material were located within the area south of Building 1818.

g. Deactivation Furnace (Area 4000) was located on an 11.2-acre site within the Area 4000. All types of small arms, ammunition components, and tracers were destroyed in the furnace by burnout.

h. Normal Maintenance Area (Area 8000) consisted of 7.8 acres and provided additional facilities for the maintenance, modification, renovation, and demilitarization of ordnance.

REMIDS Description

The airborne data collection system consists of an active/passive line scanner, real-time processing and display equipment, and navigational equipment and is described in detail elsewhere (Ballard et al. 1992). The scanner collects three channels of optically aligned image data consisting of two active laser channels (one polarized reflectance and the other total reflectance) and one passive thermal infrared channel. The source for the active laser channels is a neodymium yttrium-lithium-fluoride (Nd:YLF) laser from Laser Diode, Incorporated, WES-001 (no company model number). The laser radiates 1 W of continuous linearly polarized power at a wavelength of 1.053 μm (Miles, Castellane, and Goodson 1992). Two silicon avalanche photodiode detectors (one for each polarization channel) are used to measure the backscattered energy received by the scanner. The passive thermal infrared energy collected by the scanner is measured in the 8.5- to 14-μm band with a mercury cadmium telluride (MCT) liquid nitrogen cooled detector (Cespedes 1992). The real-time processing and display system is based on a massively parallel processor. The system has a scan rate of 350 scans per second with 710 data pixels per scan. The system was flown at two different altitudes. The low altitude (130 ft) flights were flown with a forward speed of 34 knots. This allows for the surface scan resolution to be nominally 1.9 by 1.9 in. Higher altitude (200 ft) flights were flown at a forward speed of 52 knots. This altitude and forward speed give a nominal surface scan resolution of 3.0 by 3.0 in. The information collected from the flights was used to characterize the site for

the presence of surface UXO. The system also incorporates an onboard scanner data recording system and a differential Global Positioning System. The use of Global Positioning System information allows the location of contaminated areas to be added into a Geographical Information System so the data can be overlaid on the base maps of the demonstration site. The detection is based on the remote identification of surface anomalies and materials that indicate the presence of surface UXO contamination. A cut-away diagram of the scanner is shown in Figure 2.

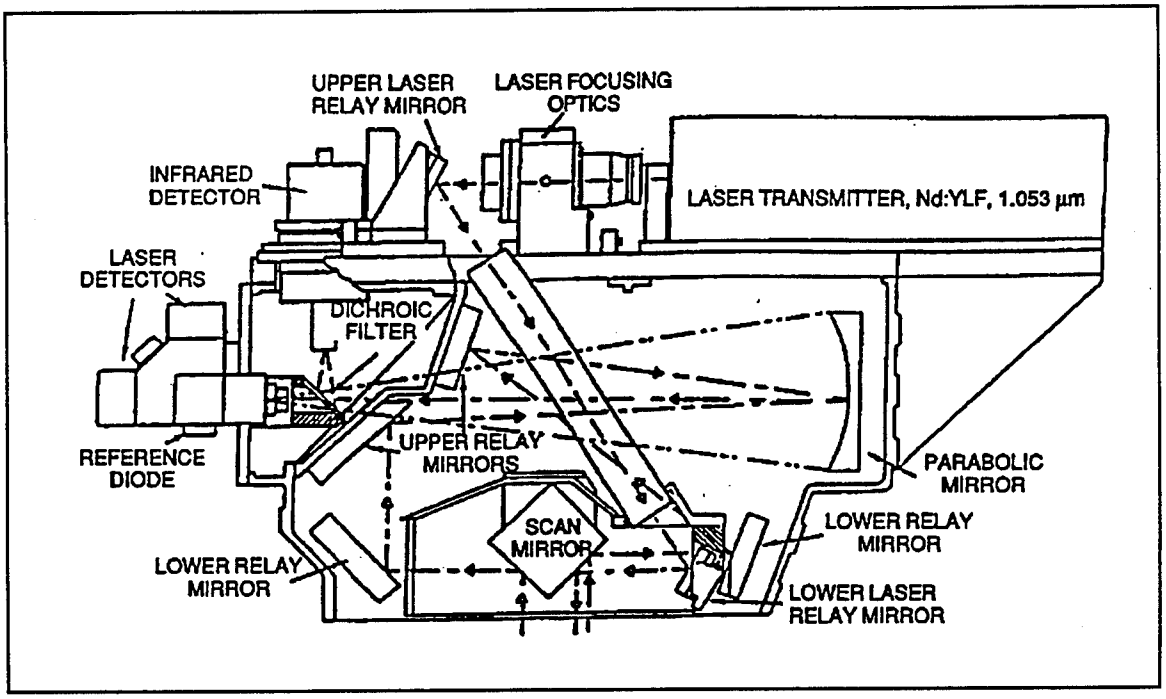


Figure 2. Scanner physical and optical layout

3 Site Preparation and Data Collection

Base Map and Bench Marks

The Huntsville Division furnished the base map file and the bench mark file for Black Hills Army Depot. The bench marks are given in NAD-27 lat/lon and zone 4002 state plane. Huntsville Division personnel generated the base map file for the contamination map overlays from drawings drafted in 1959. Seven bench marks, PT04, PT09, PT16, PT17, PT20, PT21, and PT25, were used to calculate the registration error of the base map. The RMS error (Environmental Systems Research Institute, Inc. 1992) from the registration of the base map was 190 ft. The larger registration errors were at the perimeter of the base map. The registration errors at the base station and calibration site were 9 ft. The seven bench marks used for registration were distributed throughout the entire facility.

Calibration Site

The calibration site consisted of the water containers, roofing material, surrogate mines, a surrogate hand grenade, black and white panels, and shrapnel. The surrogate mine types were the M15, M19, and RAAM. The surrogate hand grenade type was a World War II pineapple. WES personnel used the roofing material, surrogate mines, and shrapnel to calibrate the active laser sensor while hot and cold water containers and black and white panels were used to calibrate the passive infrared sensor. Apparent temperatures of the items in the calibration site were recorded using a hand-held radiometer each time a test flight flew over the calibration site. The image data of the calibration site is shown in Figure 3. The calibration site was located north of the north fence boundary of Burning Ground 2 in the vicinity of bench mark PT21.

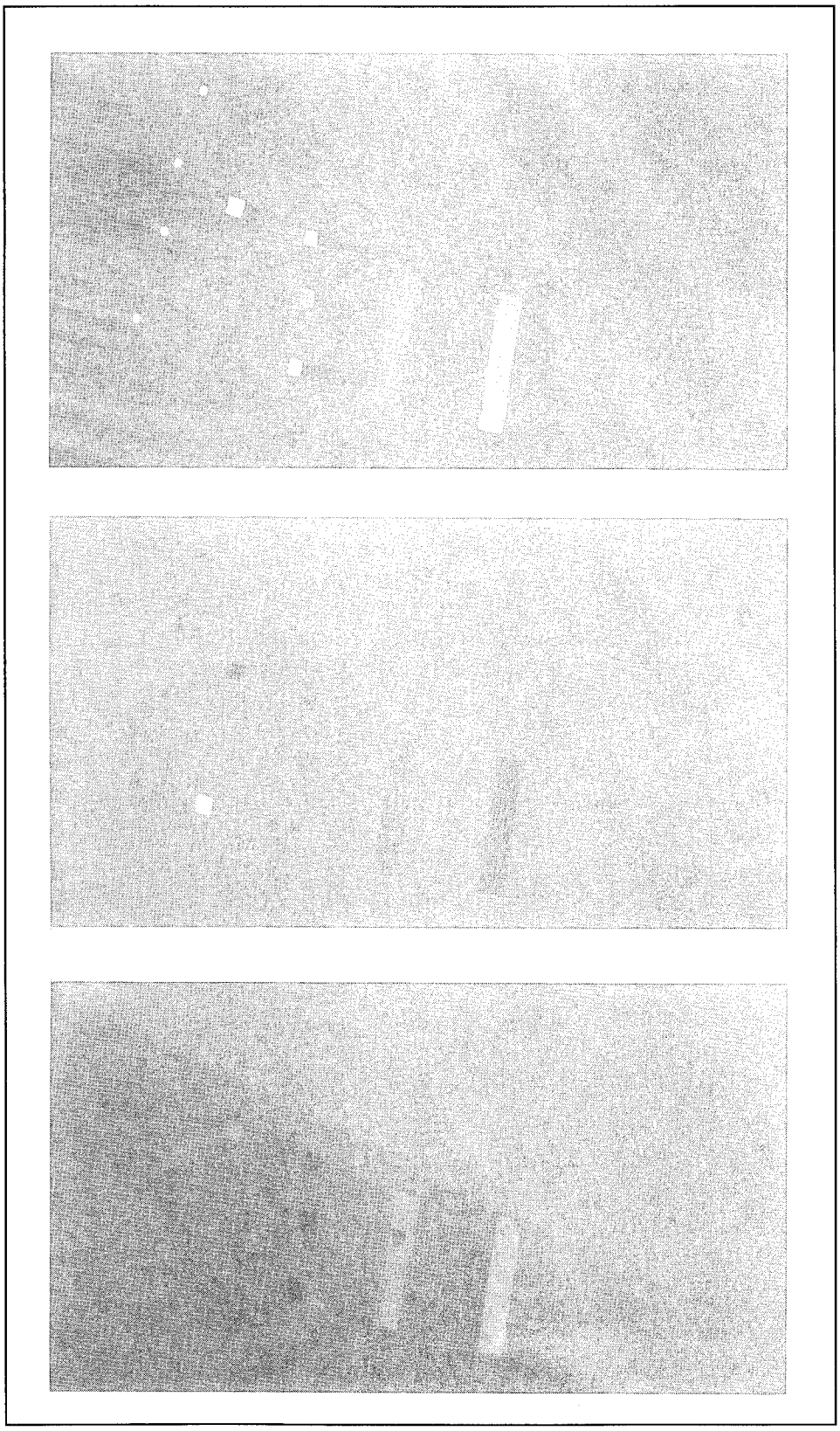


Figure 3. Image data of calibration site (polarization (top), reflectance, thermal (bottom))

Test Flights

The data collection of the sites of interest at the Black Hills Army Depot was performed in seven test flights starting on 12 July 1993, and completed on 15 July 1993. WES personnel generated a coverage map showing the paths of the test flights flown over the depot. Test flights 1, 2, 3, and 4 were flown at an altitude of 130 ft. This altitude produced a nominal swath width of 94 ft. Test flights 5, 6, and 7 were flown at an altitude of 200 ft. The nominal swath width of these flights was 145 ft. The flight paths of the seven test flights are shown in Figure 4, and the coverage map for the test flights is shown in Figure 5. Approximately 300 acres per test flight were covered for test flights 1, 2, 3, and 4; approximately 900 acres per test flight were covered for test flights 5, 6, and 7. The scanner data recorder used on these test flights was a Honeywell VLDS with a storage capacity of 5.2 GBytes per data certified VHS tape. Scanner data of 42 GBytes were collected from the test flights.

Flight paths shown in Figure 5 indicate overlapping of flight paths resulting in uncovered areas (holidays) in the site of interest. This mainly took place in the burning ground sites and was due to strong winds and the lack of ground features to aid the pilots in maintaining a desired ground track. The igloo block sites had roads that were excellent ground features; therefore, the problem with overlapping flight paths in these sites was minimal.

Global Positioning System Information

The Global Positioning System (GPS) (Trimble Navigation, Ltd. 1992) used for the survey is independent of the scanner system. The synchronization of the information between the GPS system and the scanner system is integrated together via time stamps. Huntsville Division and WES personnel set up a GPS base station at bench mark PT21, which is located southwest of the calibration site. A rover GPS unit was used during the test flights to collect the GPS location data of the helicopter. WES personnel generated differential GPS location data from the base station and the rover unit at the end of the test flights for that day. This location data were converted from NAD-83 lat/lon to NAD-27 zone 4002 state plane. The GPS state plane information was used to overlay the locations of the suspected contamination on the state plane base map furnished by the Huntsville Division.

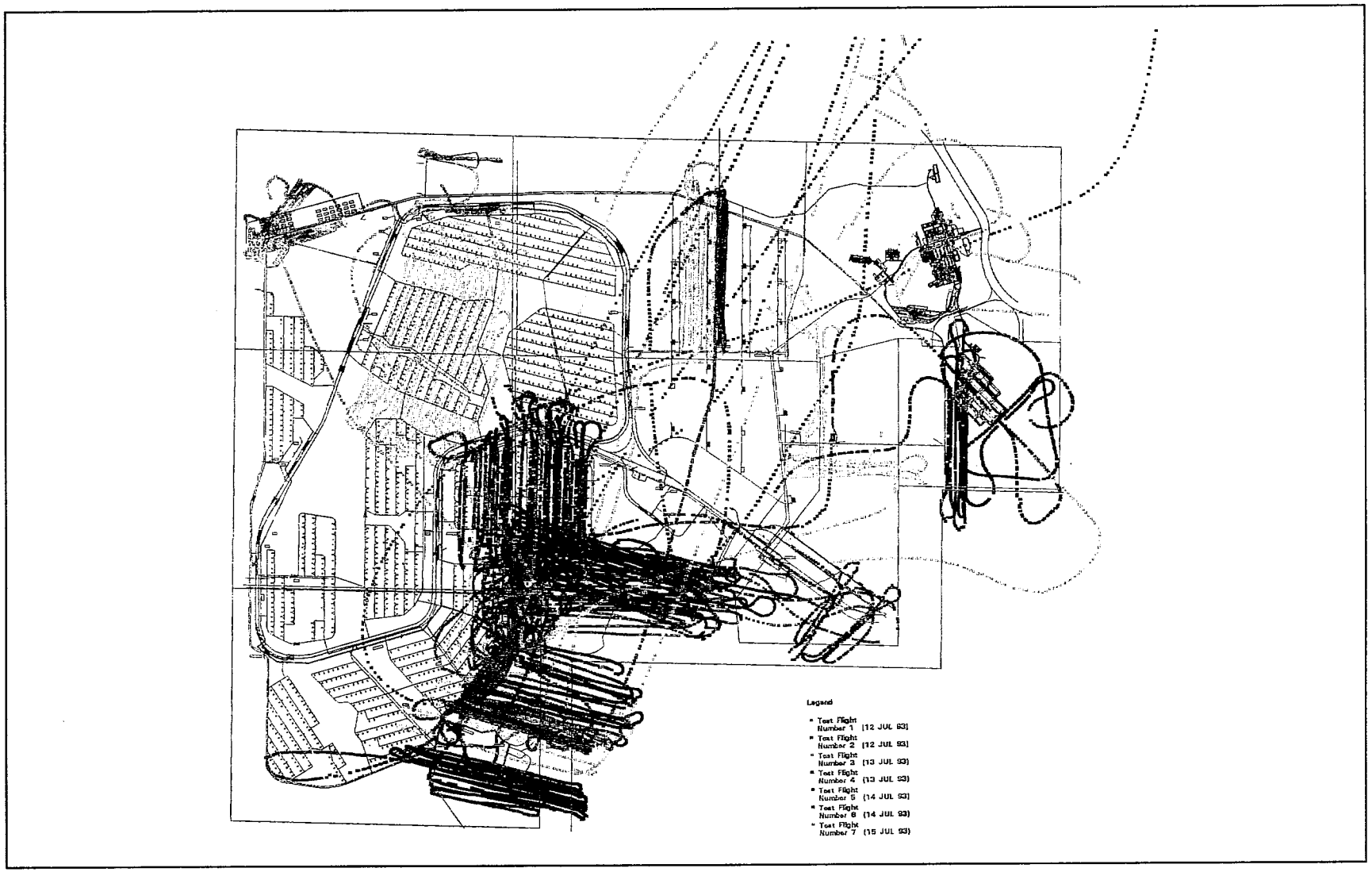


Figure 4. Test flight paths

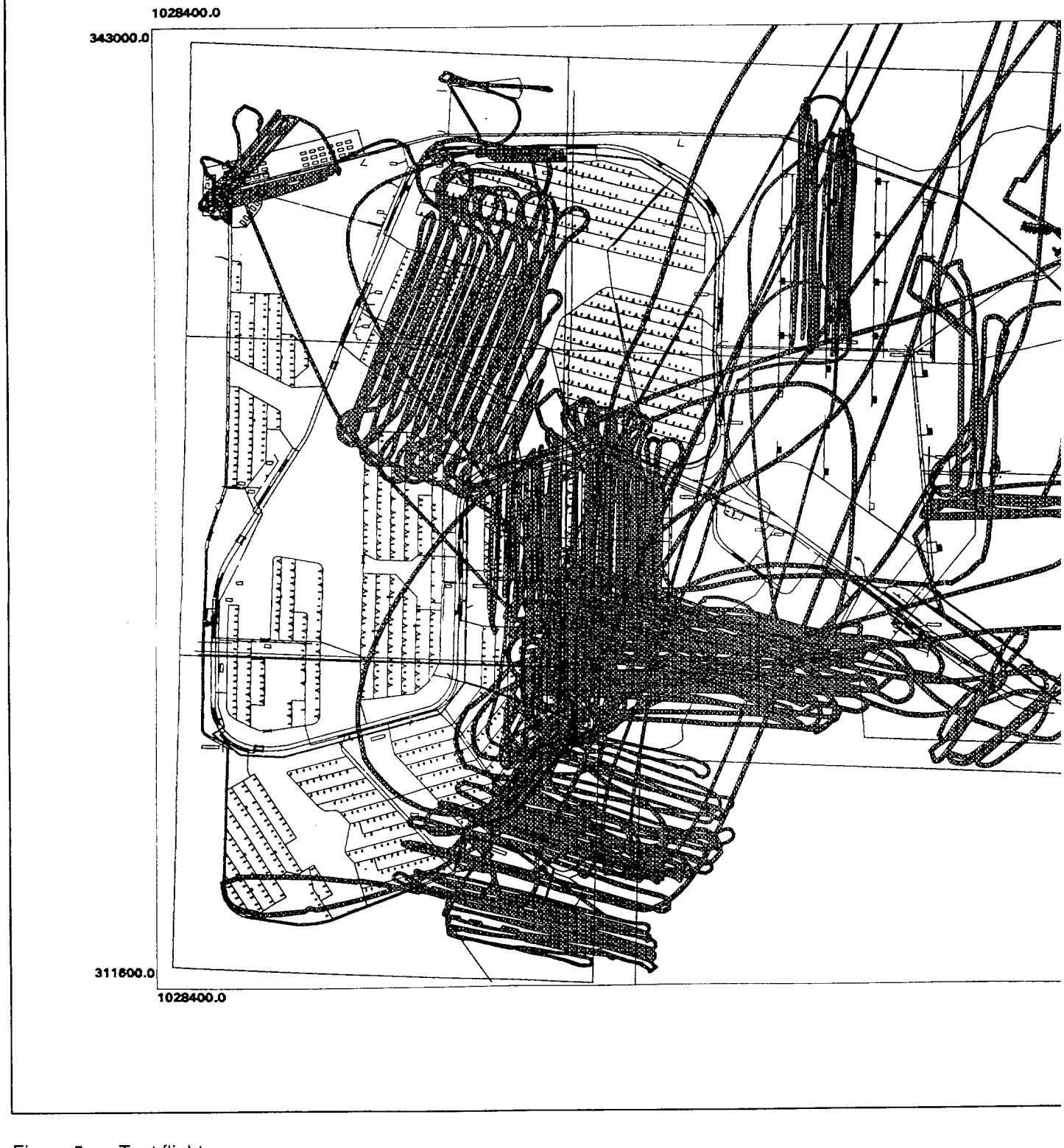
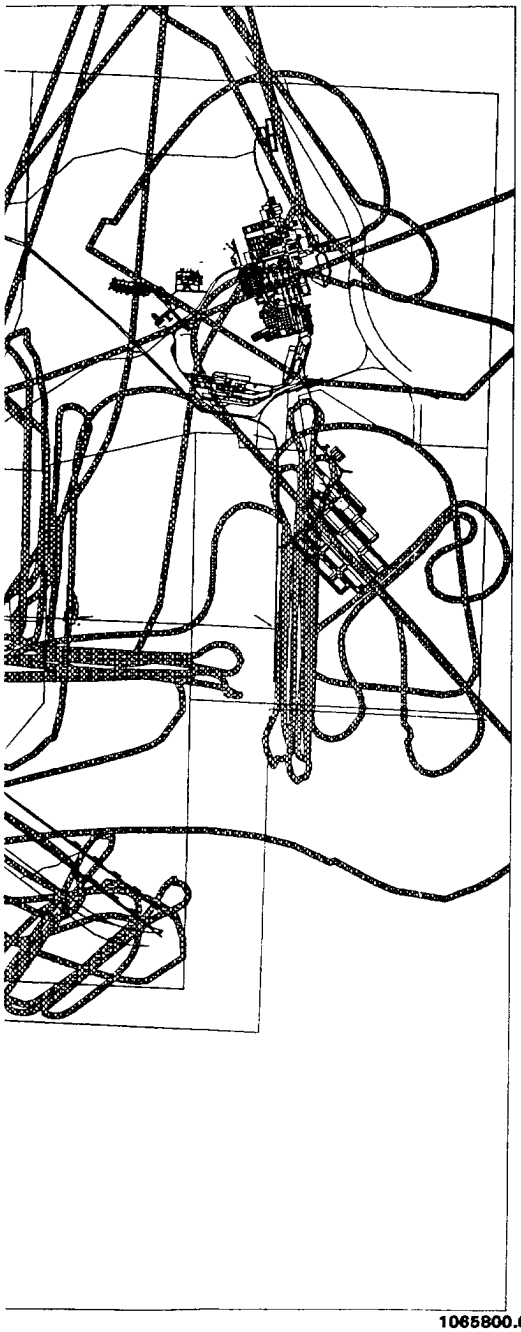


Figure 5. Test flight coverage map



Test Flight Coverage Map

Release Date: 03 Dec 1993

4000 0 4000 8000 Feet

Legend

Test Flight Coverage

Synchronization of GPS and Scanner

The following method allows for the detection of differences between GPS time stamp and the scanner time stamp. A one-tenth of a second difference in a system with a constant forward ground speed of 52 knots would cause a 9-ft offset from the actual location and the reported location. The alignment of a known ground feature is used to detect this problem. The feature is overflown at least twice at different headings. The ideal difference between the passes is 180 deg. A road feature is used in the following example to demonstrate the process. The system is flown over the road perpendicular to the center line. Four passes were made over the road, with an offset length of the 80-ft swath width along the center line of the road. Figure 6a shows a plot of the four passes with a three-tenths of a second difference between the GPS and the scanner. The misalignment of the road segments is obvious. Correcting the offset gives the following plot in Figure 6b. Fence lines and buildings are other features that can be used in this procedure.

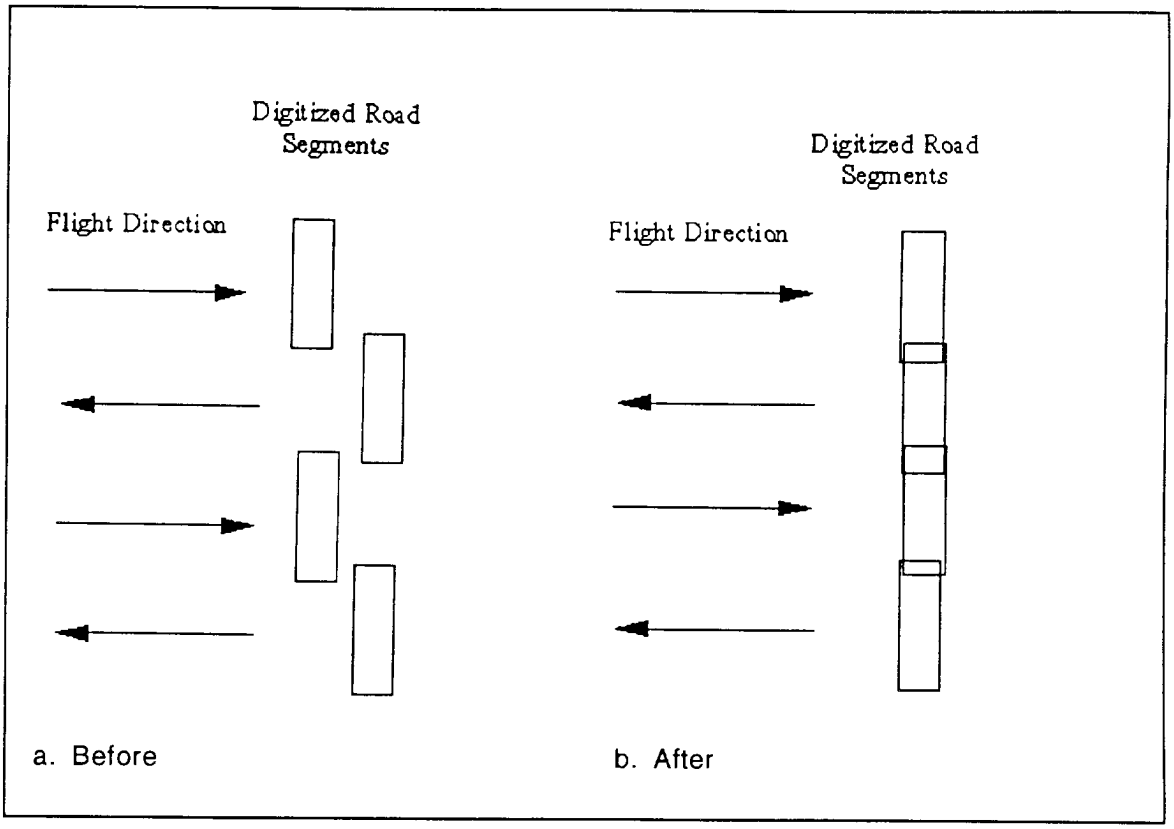


Figure 6. GPS and scanner time synchronization

4 Results

Contamination Maps

Shrapnel in the calibration site had a high-polarization return and a low-reflectance return. Image processing personnel at WES set thresholds in the image processing software to cue pixels in the imagery that aided in the visual interpretation of the data. With these set thresholds, the image processing software was also cuing water. The operator would use the thermal channel to see if the cued target was cold. If the cued target was cold, the target was identified as water and was not marked. Huntsville Division and WES personnel used the following items in the legend for the contamination maps. The single target is the location of a single target marked by the operator. The light-density target area is an area that has 10 to 20 single targets in an 80- by 80-ft area. A high-density target area is one that has more than 20 single targets in this field. A trench area is an area that shows terrain characteristics indicating that a disposal dumping trench once existed in this area. A surface anomaly is used to indicate an area where terrain anomalies, such as a denuded area because of trinitrotoluene (TNT) contamination, are present. A scattered debris area is an area that is typically close to a building or structure and contains materials that may have come from the building or structure.

Thirteen contamination maps were defined to cover the areas of interest. The coverage of the maps is shown in Figure 7. The full-size contamination maps that were made for the Huntsville Division have a scale of 1 in. = 200 ft. WES working-size maps have a scale of 1 in. = 800 ft. The complete set of the WES working-size contamination maps is given in Appendix A. During the test flights, the GPS satellite availability would be reduced from four satellites to three satellites. Accordingly, this would increase the mean distance error from 15 to 60 ft for those GPS data points. A second set of the 13 maps was generated for the areas where the satellite coverage was due to only three satellites. During test flight number 2, the satellite coverage would drop below three satellites. The GPS data points could not be corrected. Two maps were generated that contained the uncorrectable GPS data point for test flight number 2. The complete set of contamination maps consists of 13 four-satellite coverage maps, 13 three-satellite coverage maps, and 2 uncorrectable GPS coverage maps.

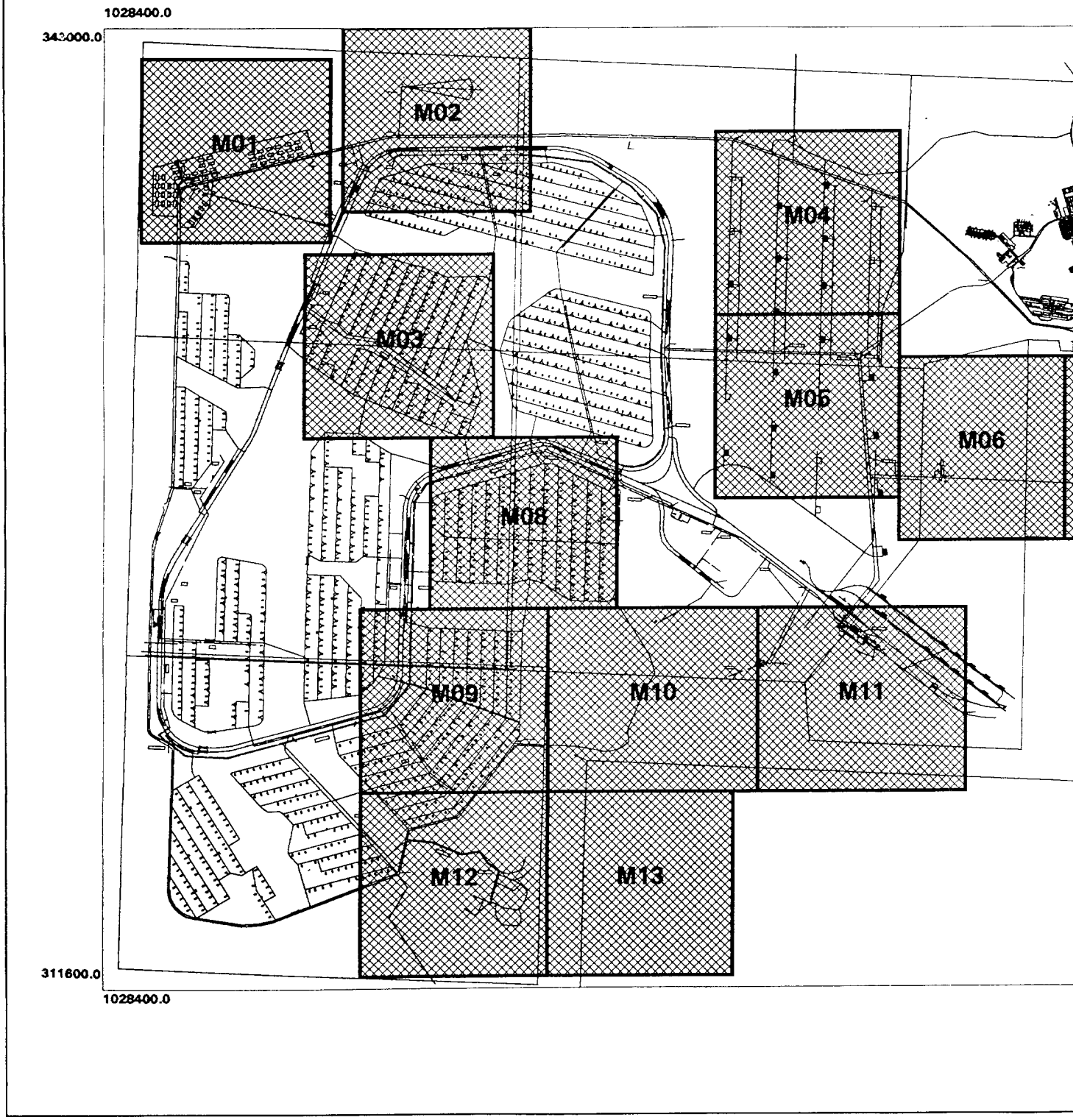
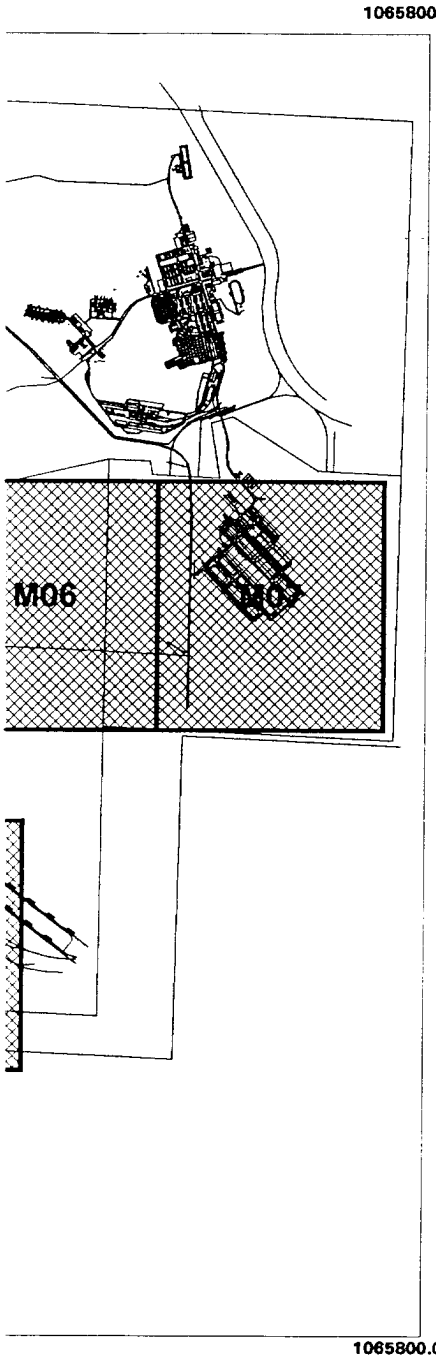


Figure 7. Contamination map coverage



Contamination Map Coverage

Release Date: 14 Oct 1994

4000 0 4000 8000 Feet

Legend

Contamination Maps

All maps were generated by the use of the ARC/INFO GIS package. The data files and macro files used to generate these maps were archived on 8mm tapes from a Sun workstation via the tar command. The macro files are listed in Appendix B.

The addition of holiday areas and digital imagery were made to the four-satellite coverage maps. The holiday areas are those areas that were not covered by the test flights as discussed in Chapter 3 and shown in Figure 5. However, when overlaying the flight paths onto the contamination maps, the holiday areas were hashed in and the flight paths were left unfilled. This is done so that the contamination features on the map are not obstructed by the hashing in of the flight paths.

The history of the map release dates are as follows:

10 Sep 1993 – Initial Release

22 Oct 1993 – Generation of second set of maps for three-satellite coverage and uncorrected GPS coverage

19 Nov 1993 – Combining of Burning Ground 2 and South of Block D data on the same map

03 Dec 1993 – Redefinition of map boundaries to avoid maps from overlaying one another

18 Aug 1994 – Addition of holiday areas and the location of digital imagery to the four-satellite coverage maps
Labeling of Contamination Maps with labels M01 through M13

14 Oct 1994 – Extended holiday areas to map boundary

The South Burning Ground 3, 3 Satellites Available, Test Flight 3 & Test Flight 4 map has no targets on it. There were no anomalies detected for the three-satellite coverage of the area. The map was added for completeness only.

Digital Imagery

Digital images of the sites were collected from the flight data tapes. The digital images are in a TGA file format (Truevision 1991). The files have the standard 18 bytes of TGA header followed by 768 bytes of color map data at the beginning of each image file. The images are screen dumps from the REMIDS image processing system and are 1,024 by 1,024 pixels with 256 colors. The filename of each image is in the format of "Mmmssc.TGA" where: mm indicates the map number, 01-13, that the image is from; ss is a serial number for the image, 01-10; and c indicates which channel, P for polarization, R for reflectance, and T for thermal. A 1,024- by 704-pixel image was cropped out of each file and hard copies were made. An example is shown in Figure 8. All of the digital imagery

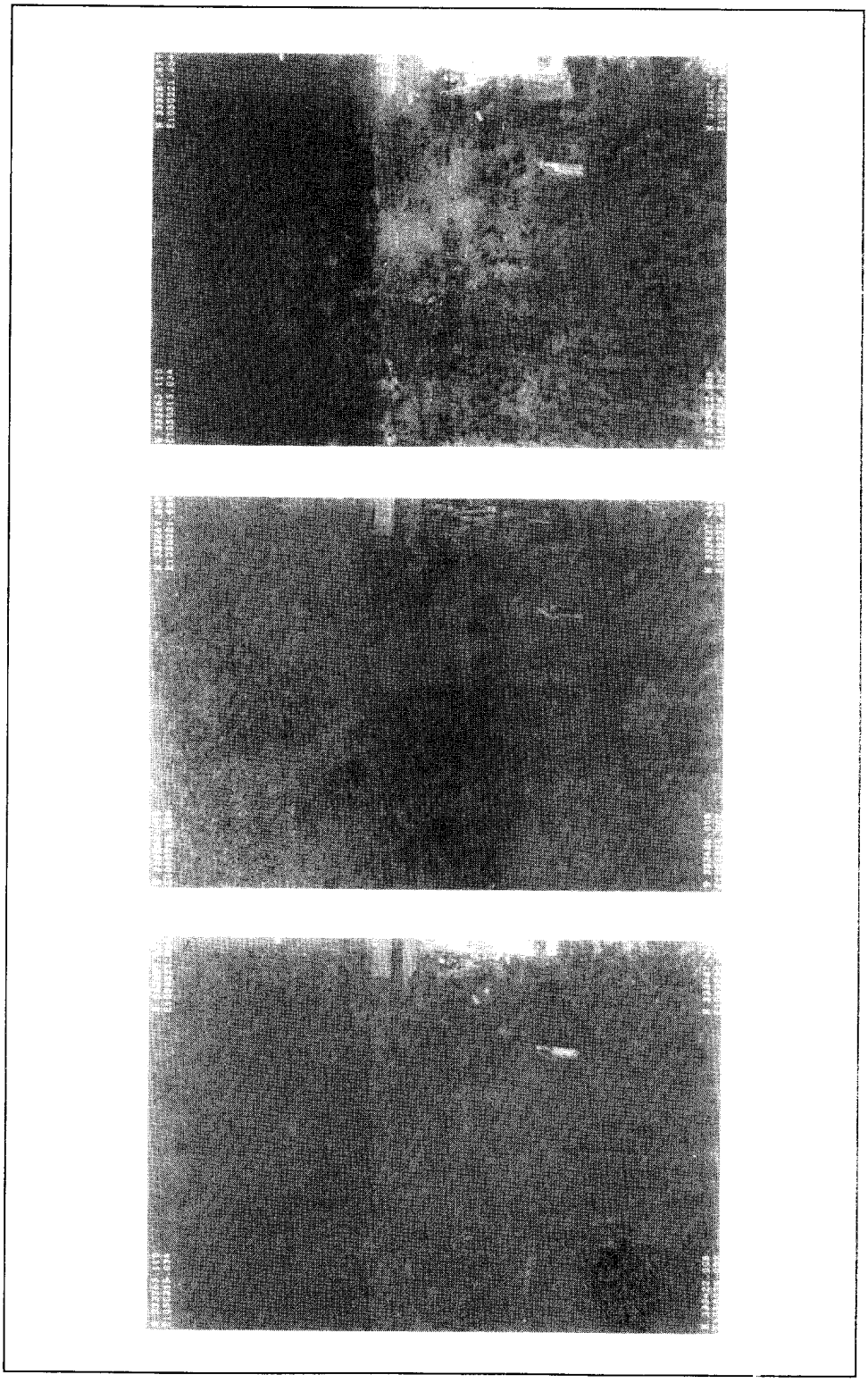


Figure 8. Digital imagery (M0506 polarization (top), reflectance, thermal (bottom))

files were archived on 8mm tapes from a Sun workstation via the tar command. The M06 Disassembly Plant Area, 4 Satellites Available, Test Flight 6 Map has very few targets on it; therefore, no digital images were generated for this map.

Archived Flight Data

The polarization, reflectance, and thermal channels were extracted from the flight data tapes and stored on CD-ROM. The CD-ROM storage specification used was the ISO-9660 format. The three channels were stored in TIFF (tag-based file format) file format (Aldus Developers Desk 1992). The housekeeping information from the flight data tapes and the GPS information were also stored on the CD-ROMs using the ASCII (American Standard Code for Information Interchange) character set with a carriage return and line feed to terminate each line.

5 Summary

REMIDS has shown the ability to detect surface anomalies that may, in fact, be surface UXO. This system has also shown the ability to detect a wide range of man-made objects. Thus, work is needed in algorithm development to classify the surface anomalies so that only those anomalies targeted are surface UXO related. The use of this scanner system over a controlled test site is recommended for the further development of the system's UXO detection algorithms. REMIDS technology is currently being enhanced by Raytheon Corporation as part of the Airborne Standoff Minefield Detection Program. Their effort will produce a downsized higher resolution active/passive system capable of operating from an unmanned aircraft. Any algorithm development for other uses of REMIDS would aid in the quick adaption of the system when the Raytheon development stage is completed.

The use of base maps with accuracy at the level of the GPS-collected coordinates of the contaminants is important. Overlaying of GPS coordinates onto an inaccurate base map could lead to a misinterpretation of the location of the surface UXO contaminants.

The need for accurate flight path control has also been shown. The elimination of overlapping flight paths and uncovered areas would be needed for manned and/or unmanned airborne platforms and obviously would increase the efficiency and reliability of the system.

References

Aldus Developers Desk. (1992). "TIFF Revision 6.0," Aldus Corporation, Seattle, WA.

Ballard, J. H., Castellane, R. M., Miles, B. H., Wesolowicz, K. G. (1992). "The REmote MInefield Detection System (REMIDS) II major components and operation," Technical Report EL-92-30, U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.

Cespedes, E. R. (1992). "Adaptive compression of multisensor image data," Technical Report EL-92-17, U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.

Environmental Systems Research Institute, Inc. (1992). *Understanding GIS: The ARC/INFO method.* Redlands, CA.

Miles, B. H., Castellane, R., and Goodson, R. A. (1992). "REmote MInefield Detection System (REMIDS) high-frequency laser output oscillation elimination," Technical Report EL-92-29, U.S. Army Engineer Waterways Experiment Station, Vicksburg, MS.

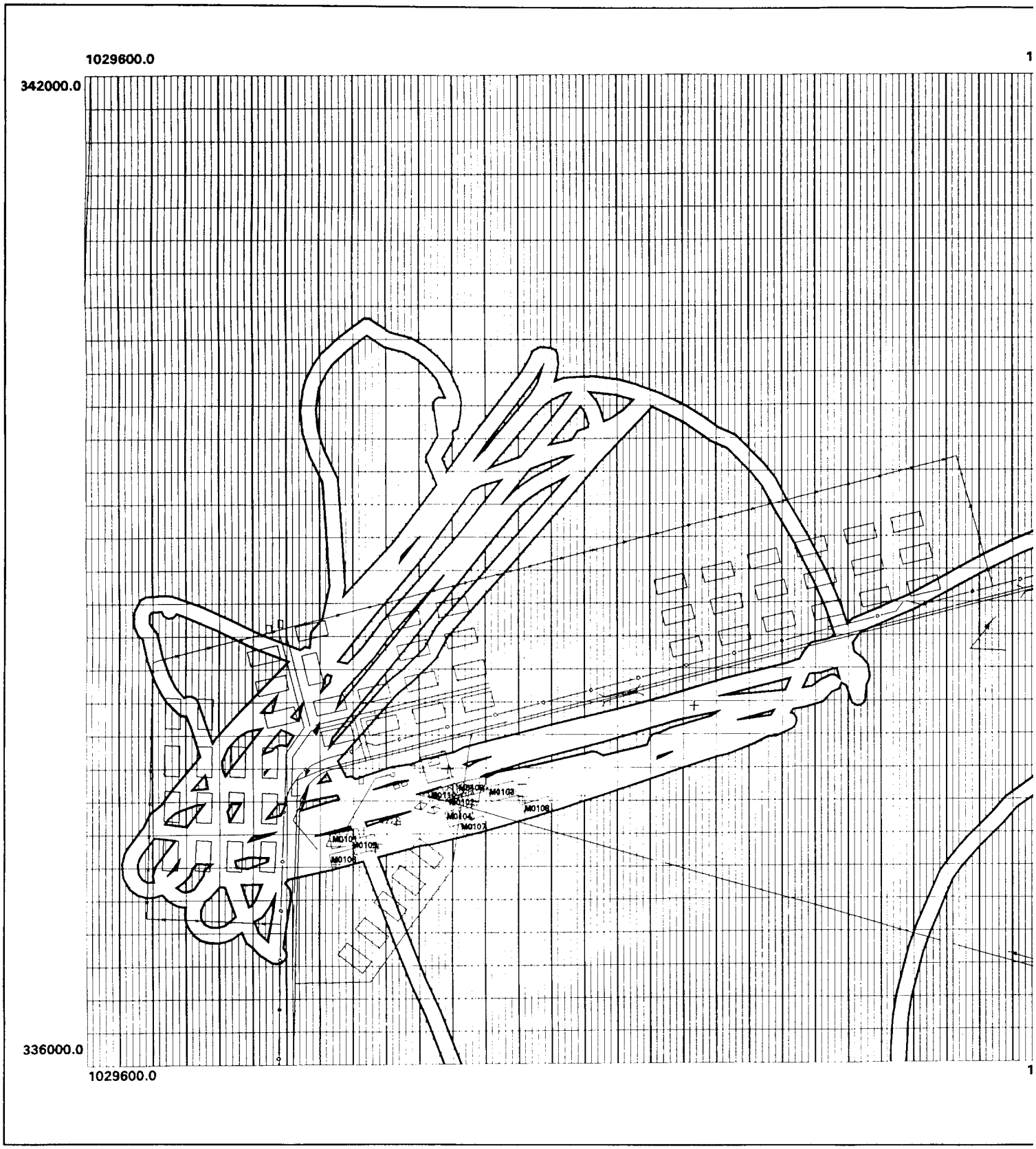
TCT-St. Louis. (1993). *Supplemental archives search report: Preliminary assessment of chemical warfare materials at the former Black Hills Army Depot, South Dakota.* Vols I, II, III, St. Louis, MO.

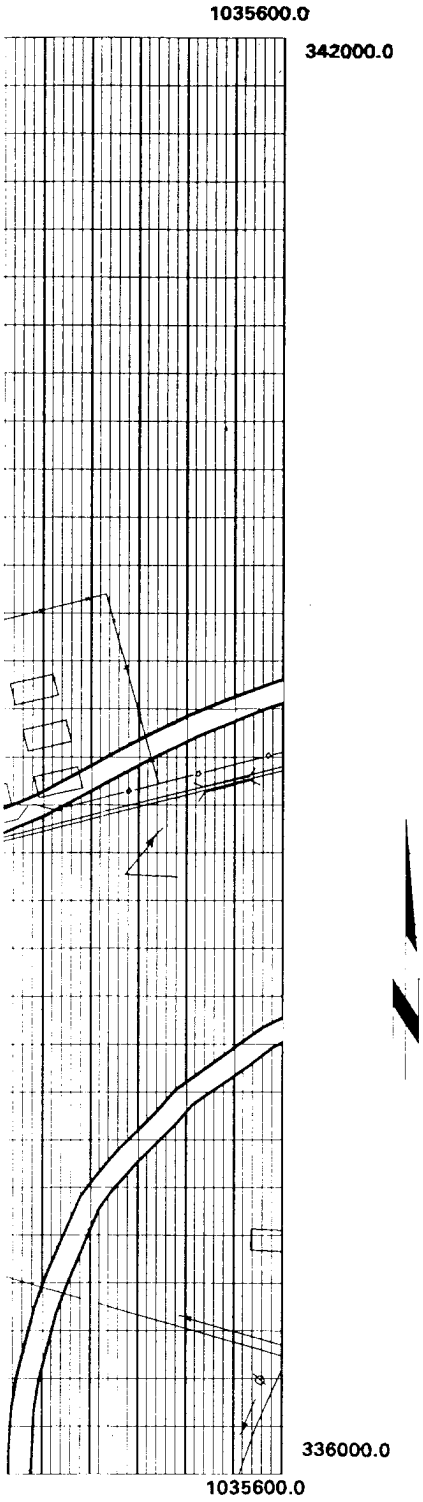
Trimble Navigation, Ltd. (1992). "System operating manuals," Sunnyvale, CA.

Truevision, Inc. (1991). "Truevision TGA file format specification," Technical Manual Version 2.2, Indianapolis, IN.

Appendix A
Contamination Maps

Thirteen contamination maps areas were defined to cover the areas of interest. The full-size contamination maps that were made for the Huntsville Division have a scale of 1 in. = 200 ft. WES working-size maps have a scale of 1 in. = 800 ft. The complete set of the WES working-size contamination maps is given in this appendix. The complete set of contamination maps consists of 13 four-satellite coverage maps, 13 three-satellite coverage maps, and 2 uncorrectable GPS coverage maps. All maps were generated by the use of the ARC/INFO GIS package. The macro files are listed in Appendix B.


1029600.0
342000.0
336000.0
1029600.0



M01 Chemical Site
4 Satellites Available
Test Flight 3

Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

Single Targets

Light Density Target Areas

Holiday Areas

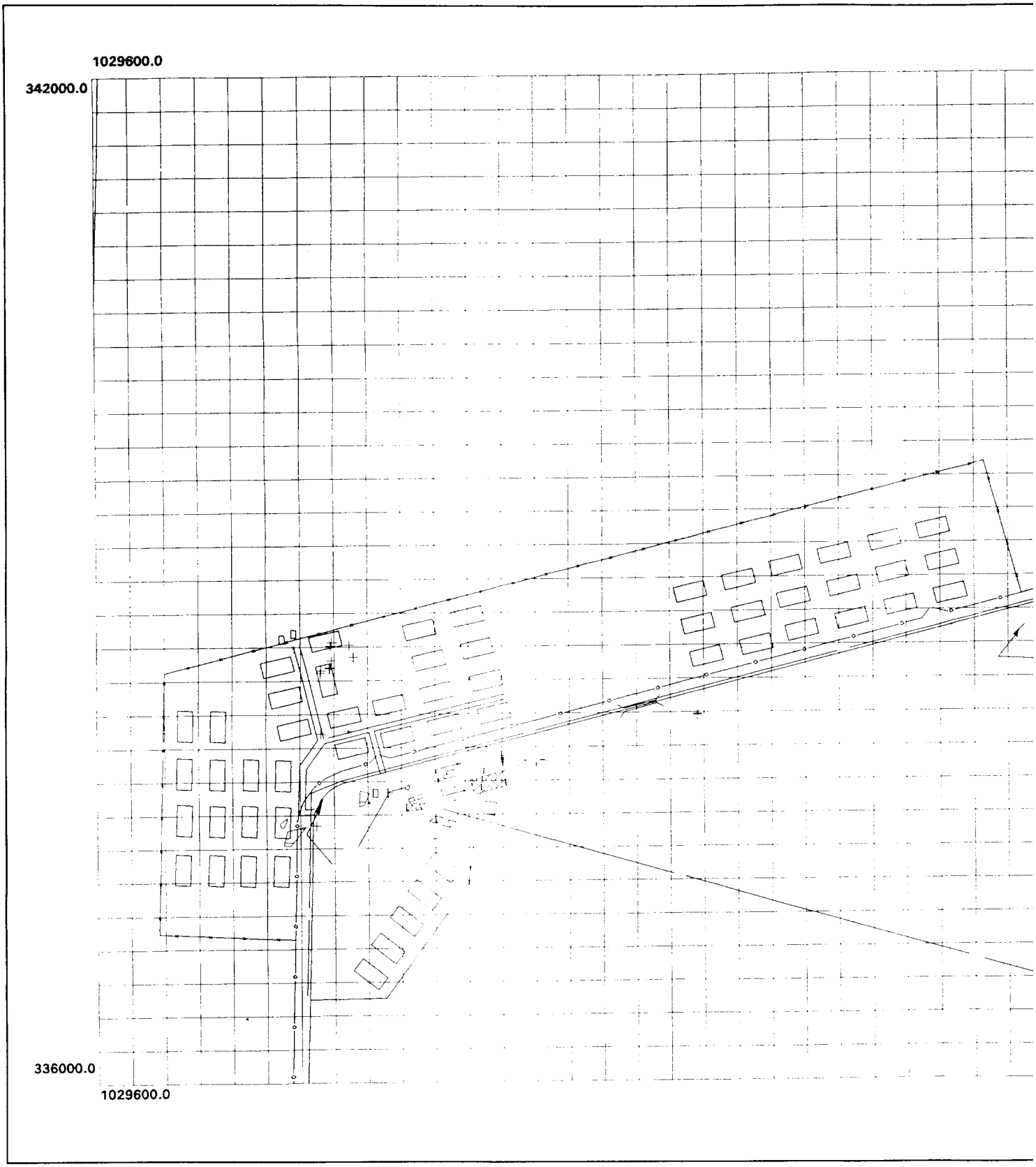
Trench Areas

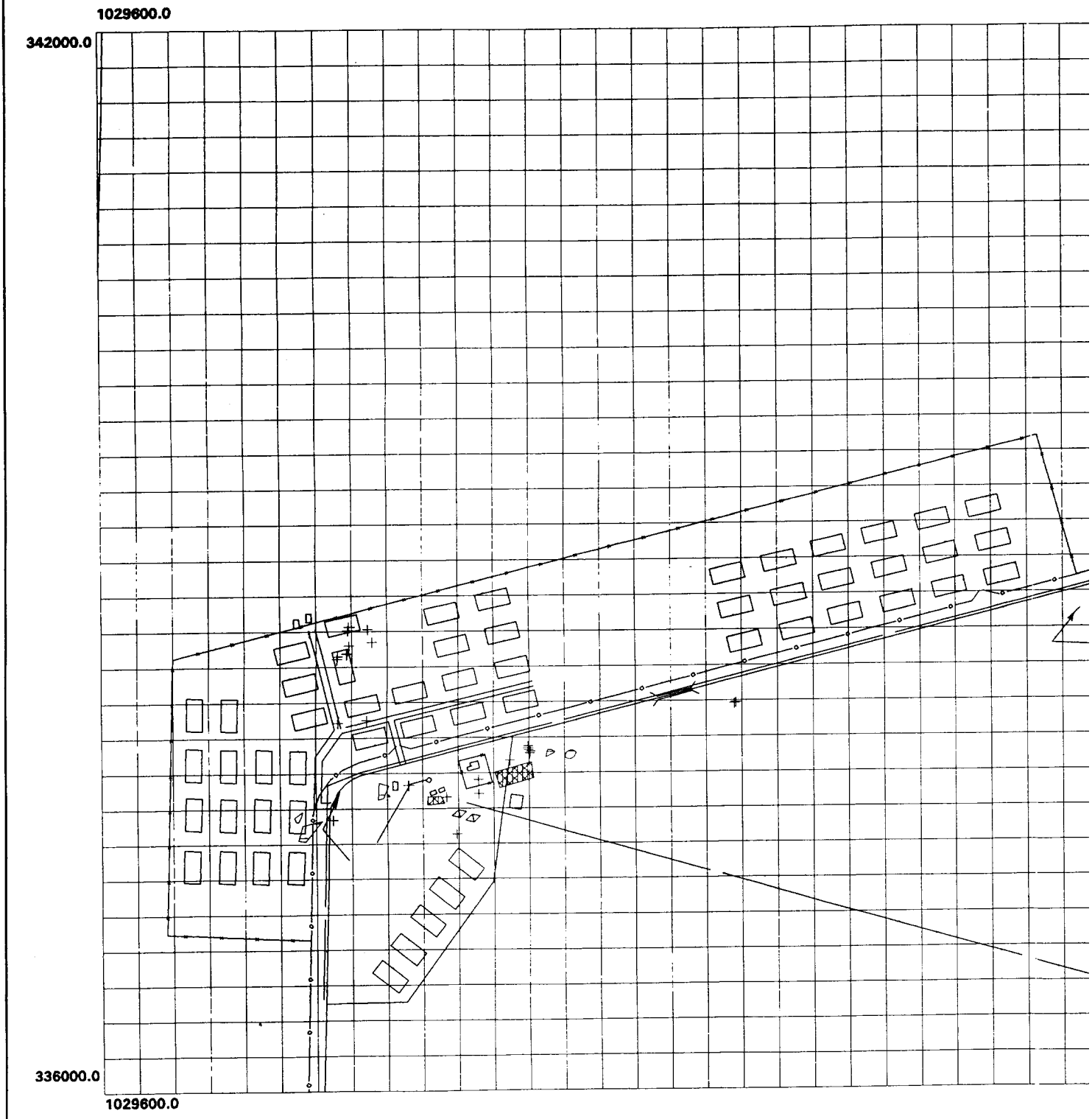
Digital Imagery

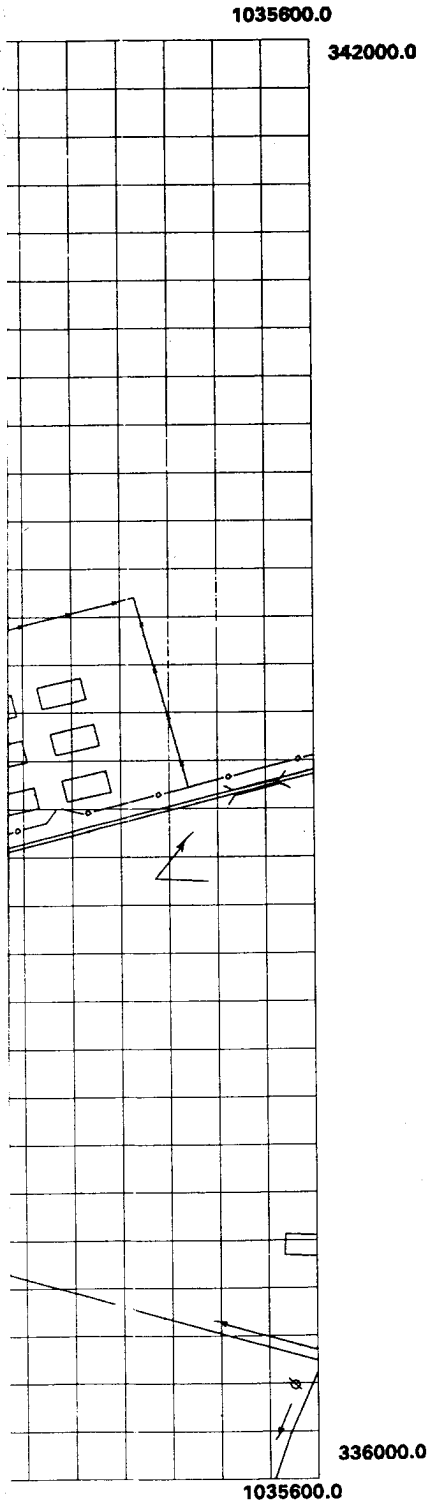
Surface Anomalies

Scattered Debris

High Density Target Areas

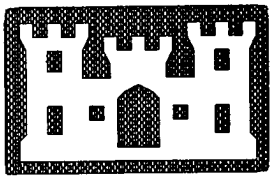

1029600.0
342000.0
336000.0
1029600.0

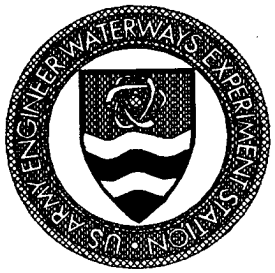

1029600.0
342000.0
336000.0
1029600.0

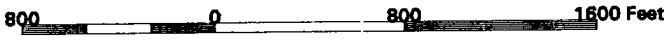


Chemical Site
3 Satellites Available
Test Flight 3

Release Date:
03 Dec 1993

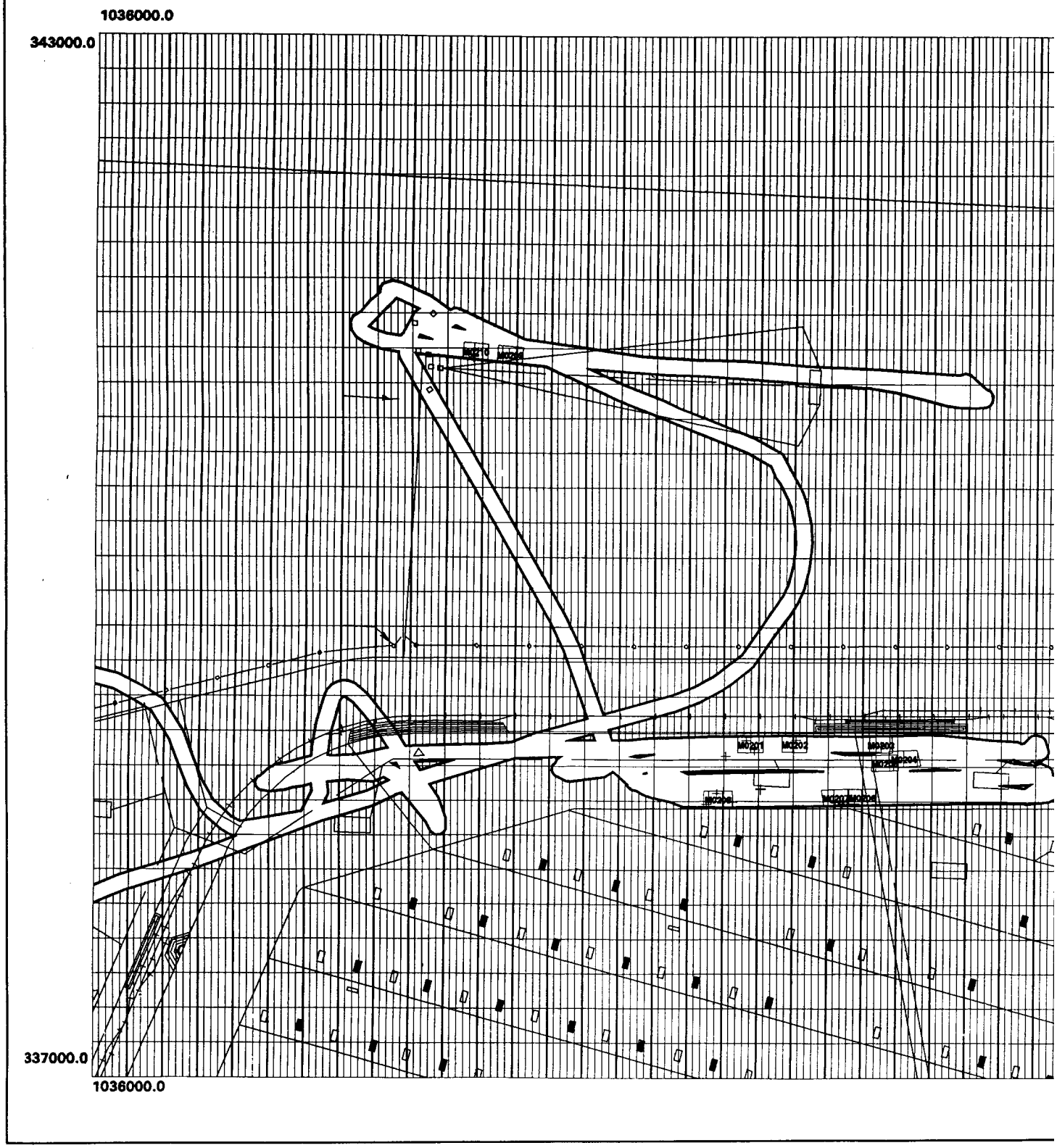


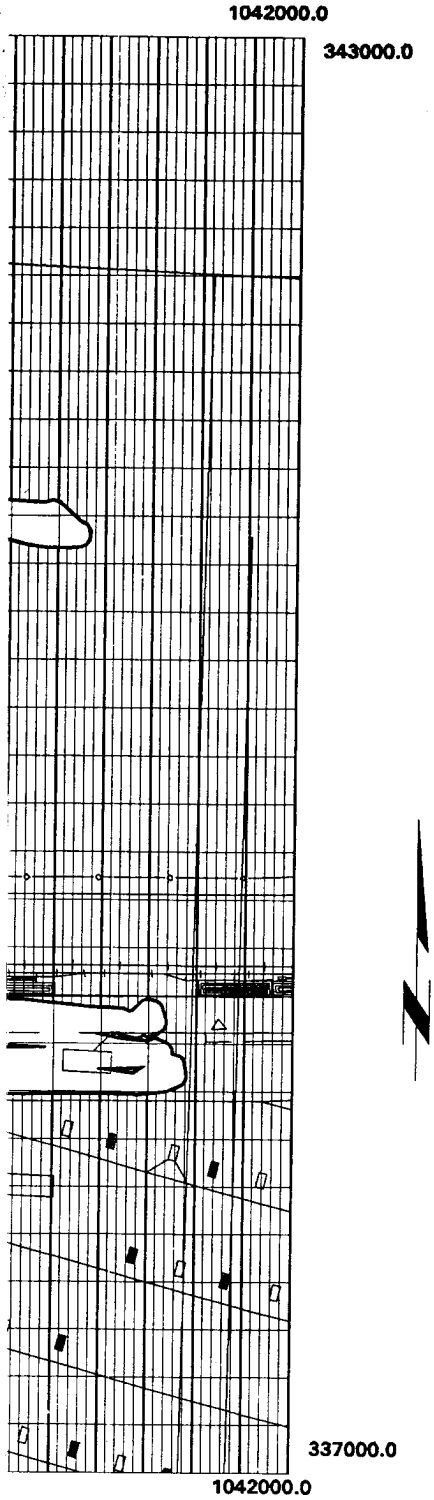




Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1036000.0
343000.0
337000.0
1036000.0



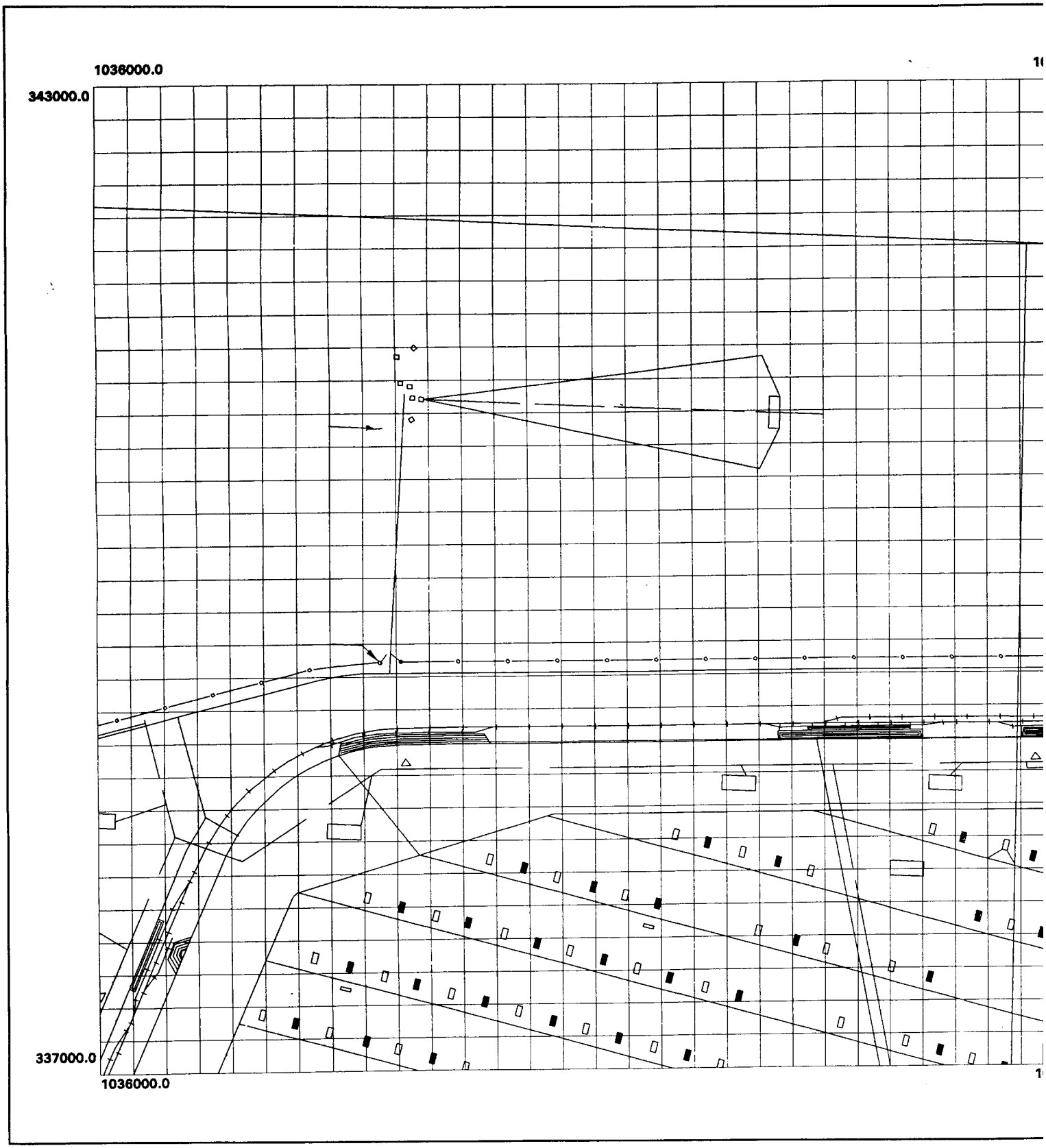
M02 Target Range
4 Satellites Available
Test Flight 3

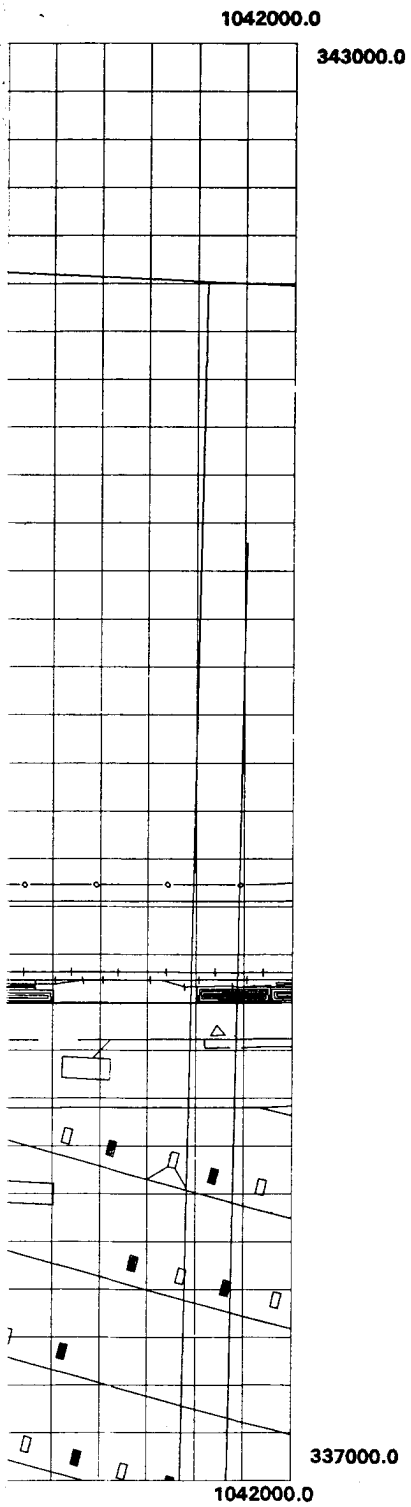
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Holiday Areas
- Trench Areas
- Digital Imagery
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1036000.0
343000.0
337000.0
1036000.0



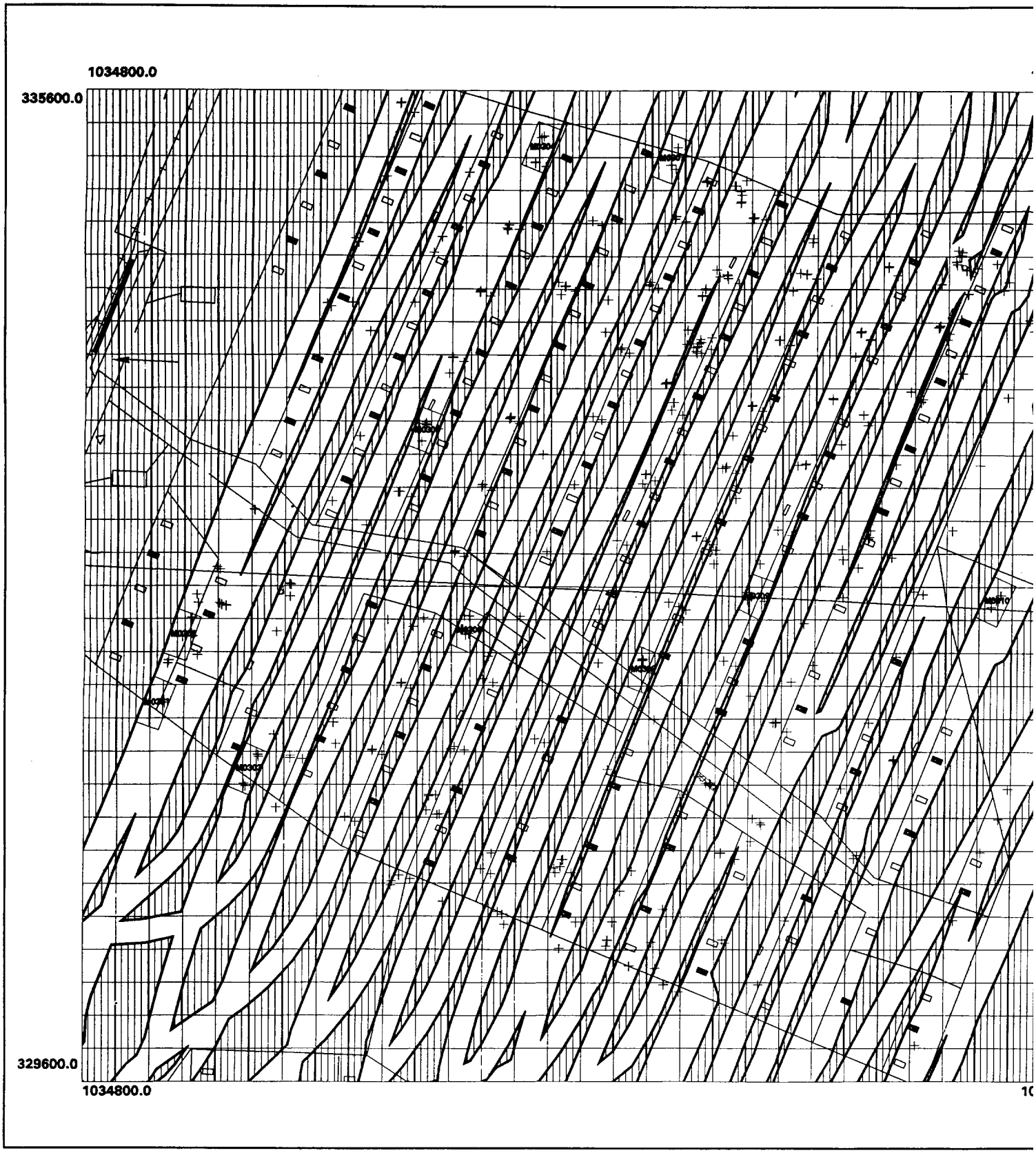
Target Range
3 Satellites Available
Test Flight 3

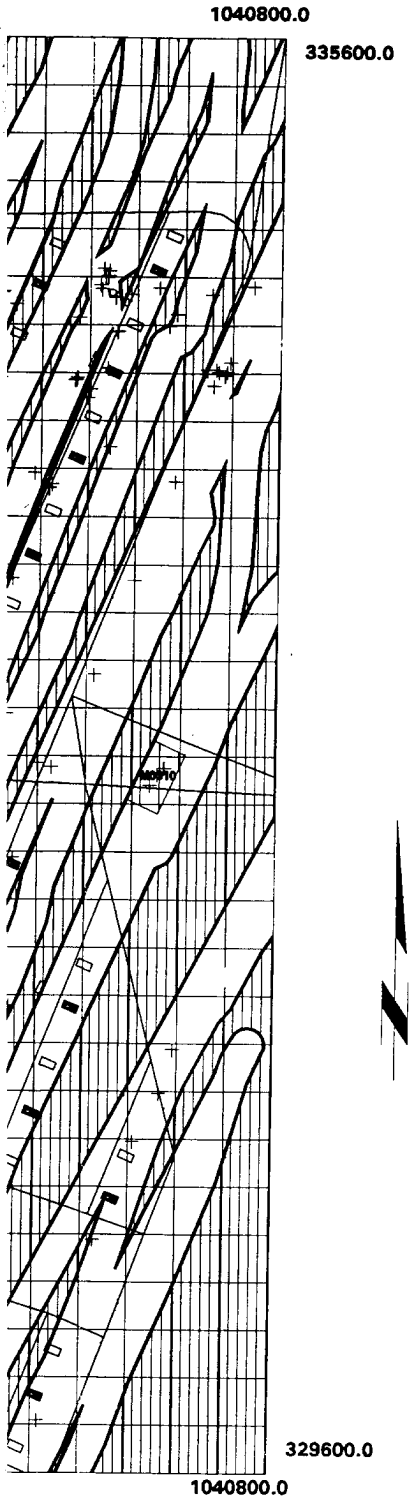
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1034800.0
335600.0
329600.0
1034800.0



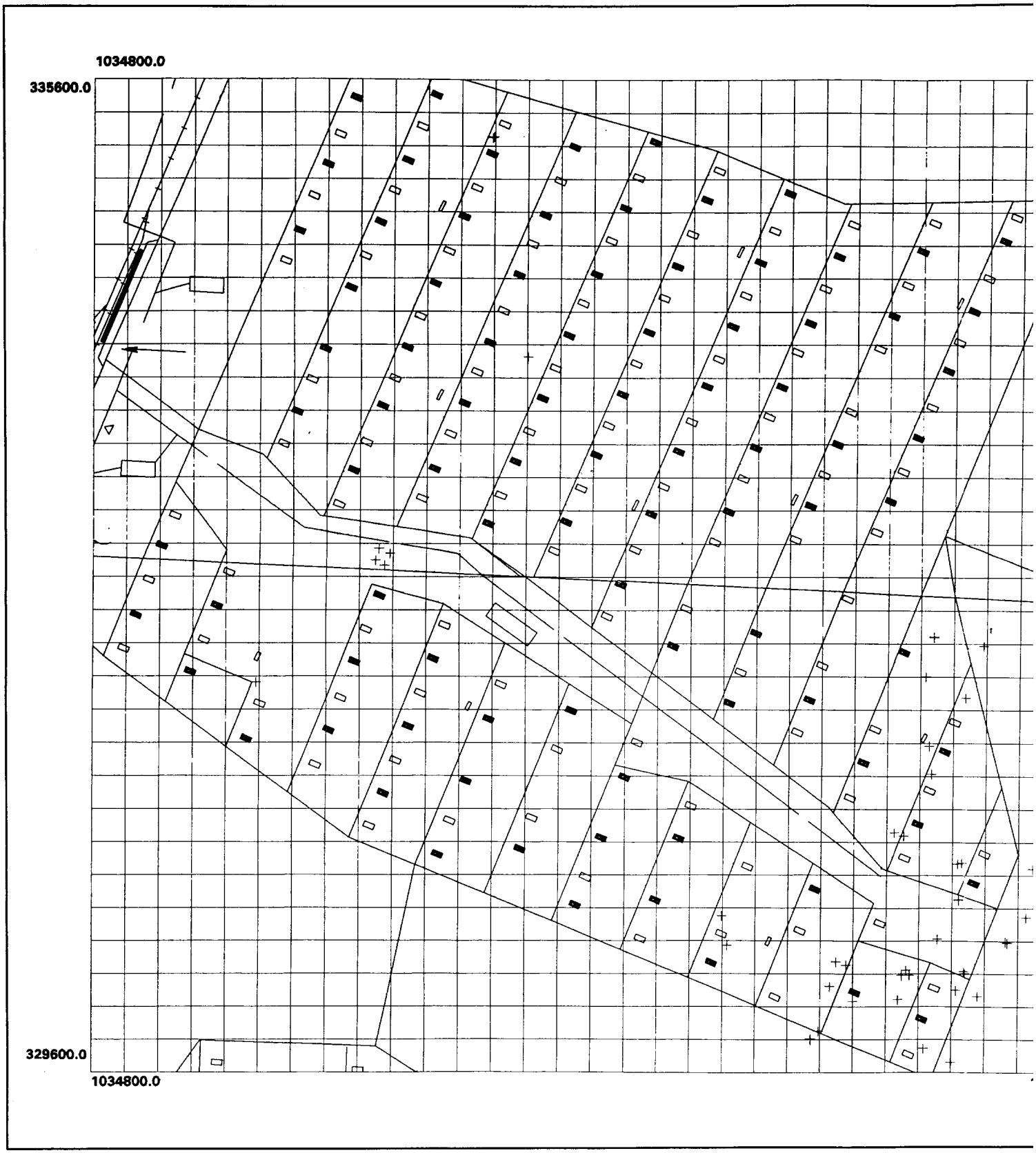
M03 Block G
4 Satellites Available
Test Flight 7

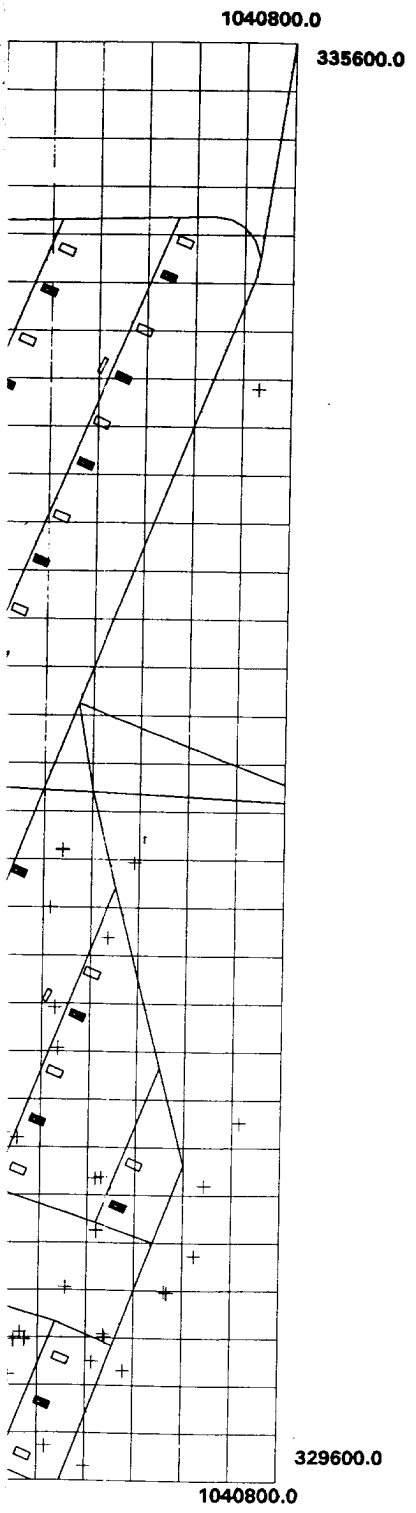
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery


1034800.0
335600.0
329600.0
1034800.0



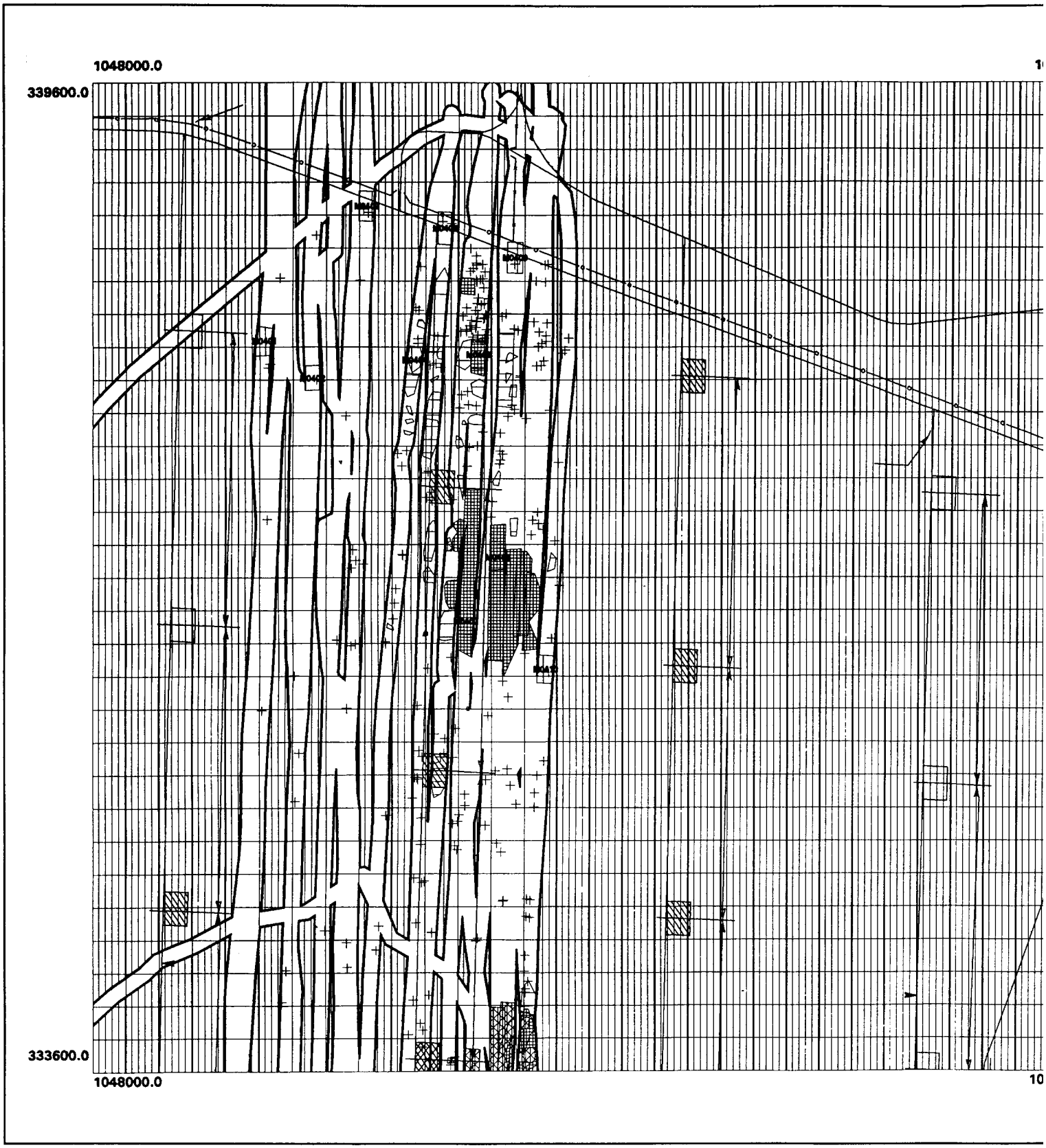
Block G
3 Satellites Available
Test Flight 7

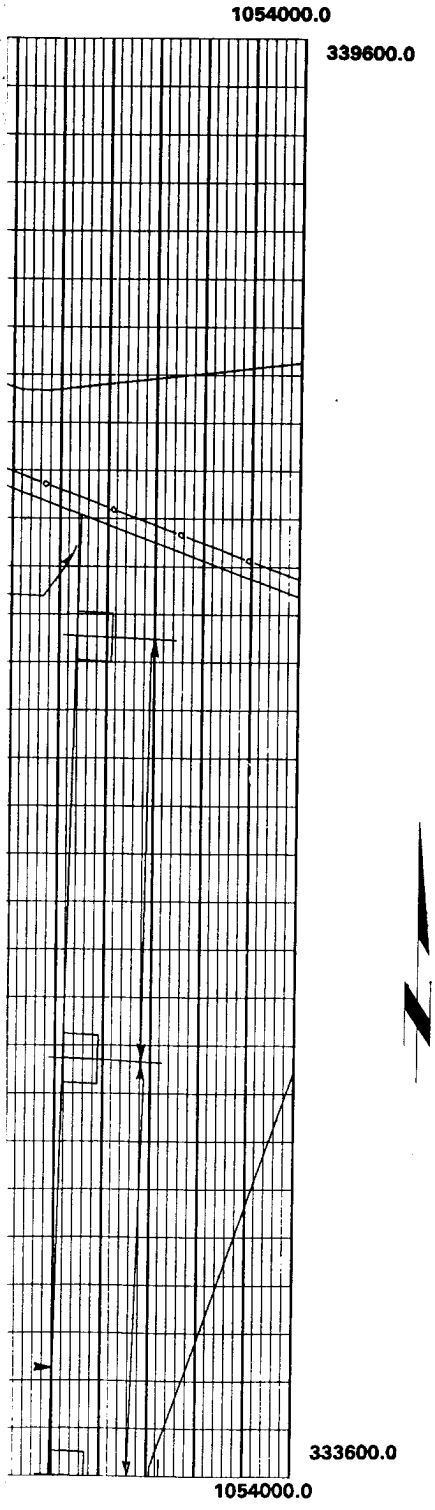
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1048000.0
339600.0
333600.0
1048000.0



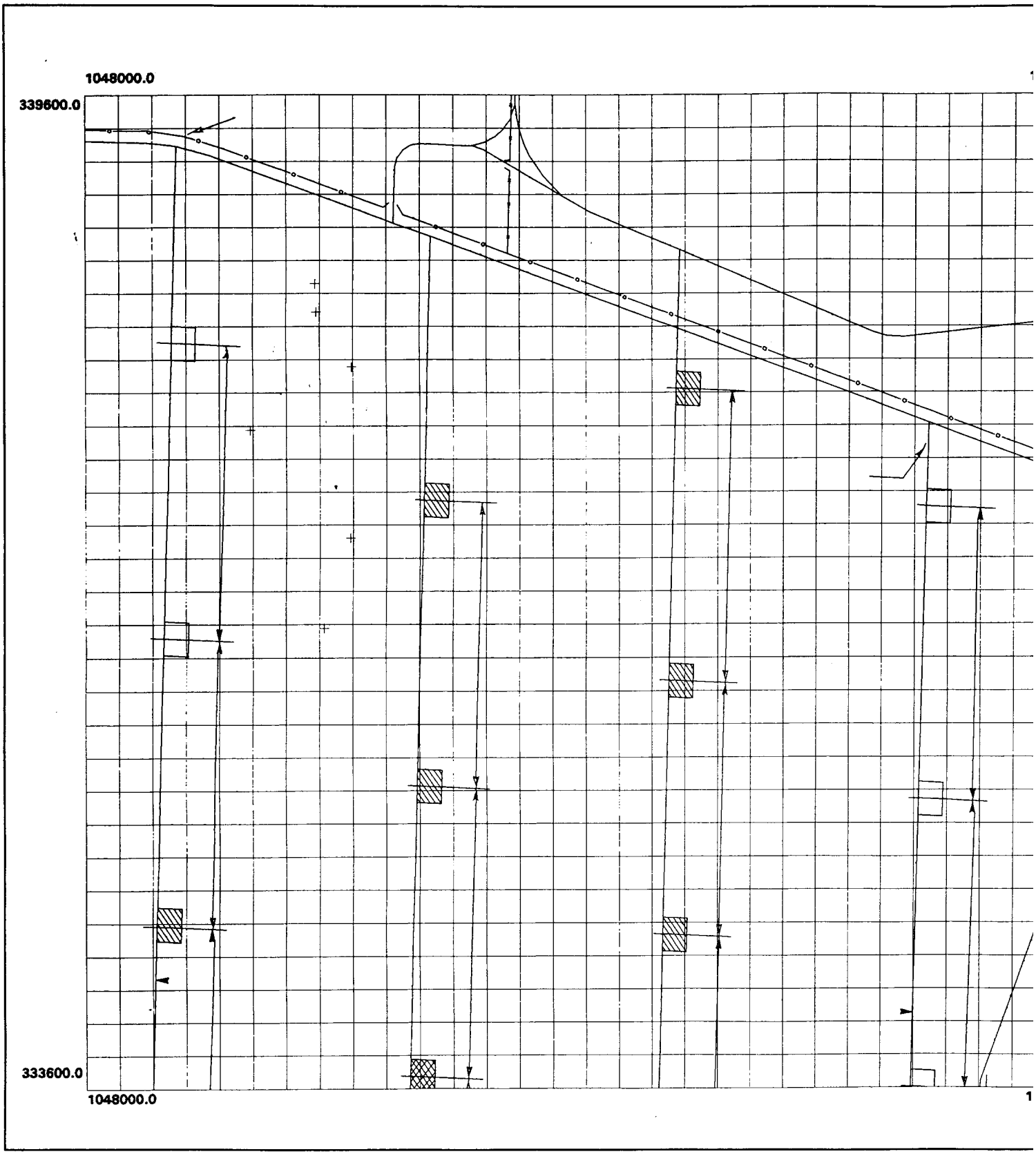
M04 Burning Ground 3
4 Satellites Available
Test Flight 3 &
Test Flight 4

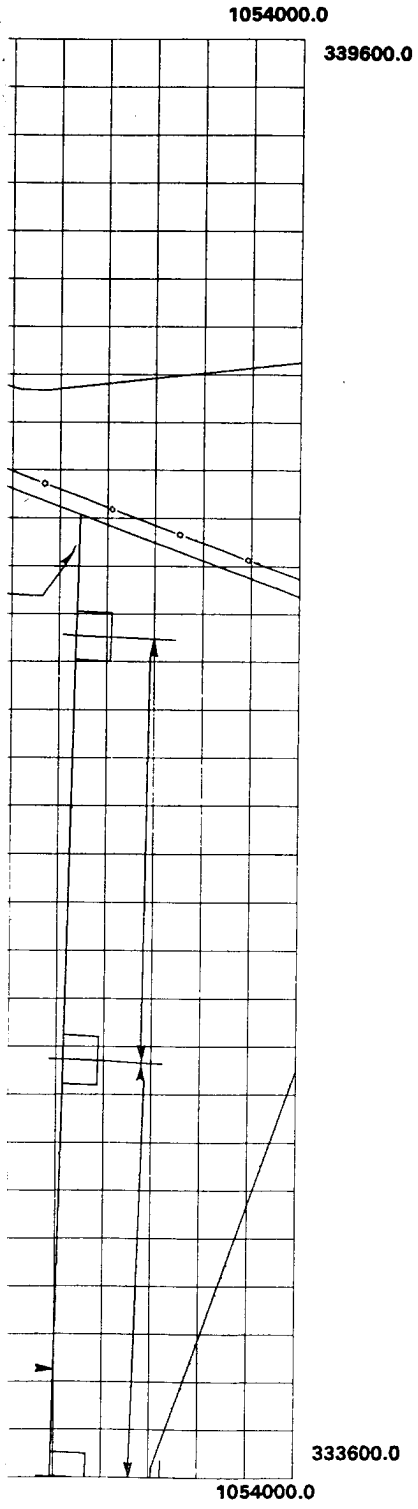
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery


1048000.0
339600.0
333600.0
1048000.0



Burning Ground 3
3 Satellites Available
Test Flight 3 &
Test Flight 4

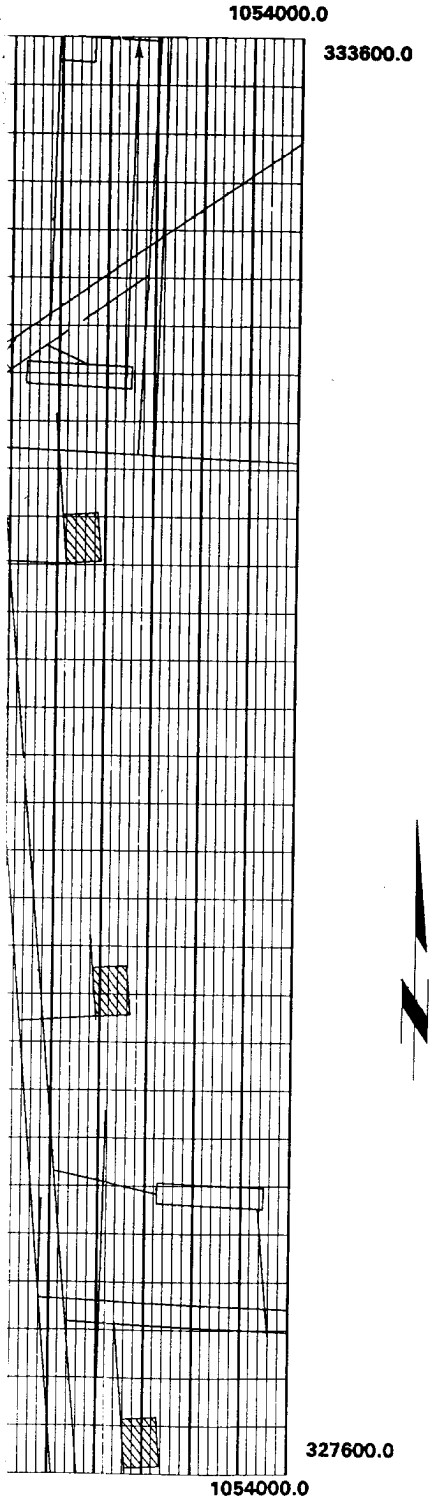
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- **Single Targets**
- **Light Density Target Areas**
- **Trench Areas**
- **Surface Anomalies**
- **Scattered Debris**
- **High Density Target Areas**

1048000.0
333600.0
327600.0
1048000.0



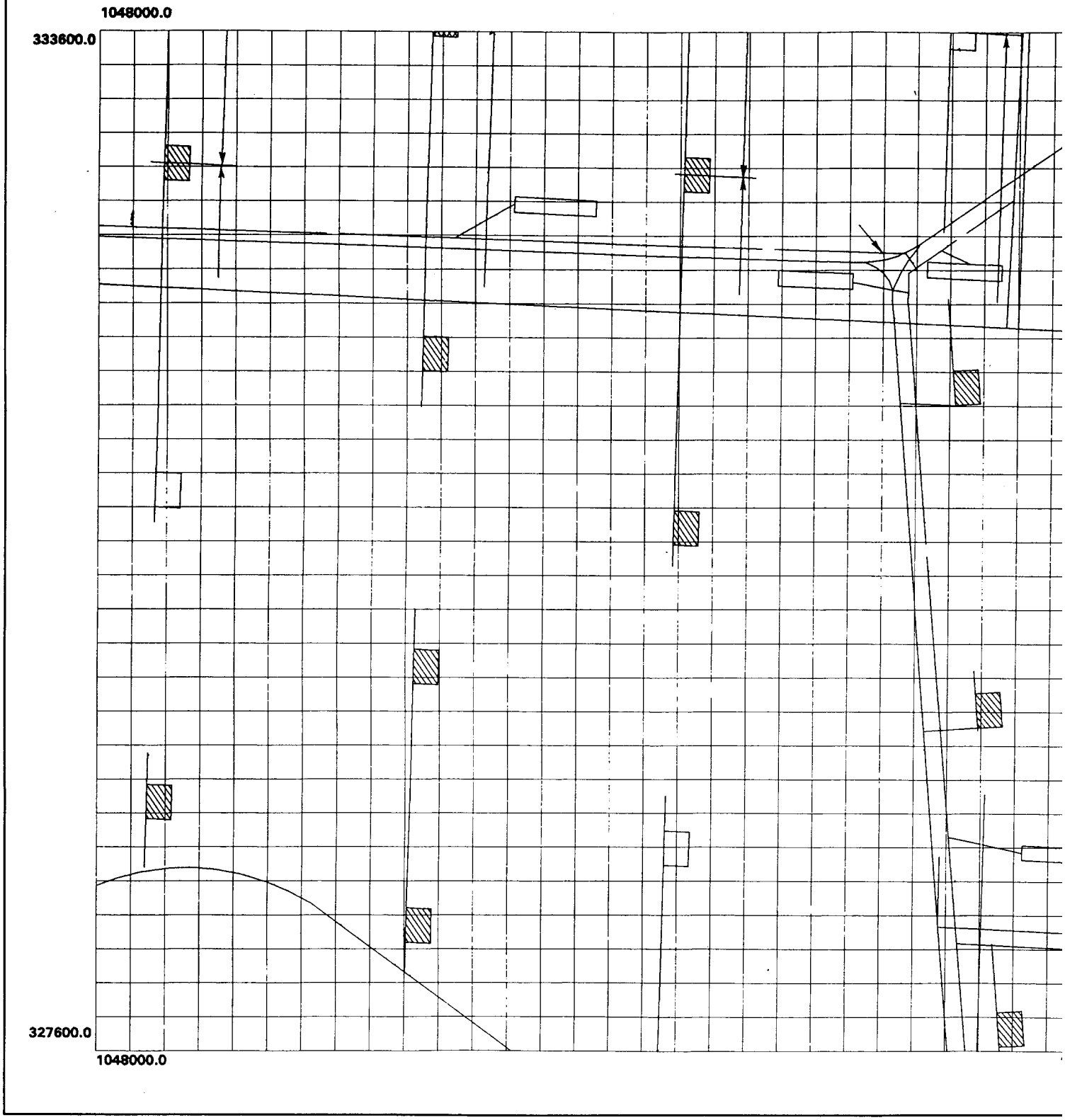
M05 South Burning Ground 3
4 Satellites Available
Test Flight 3 &
Test Flight 4

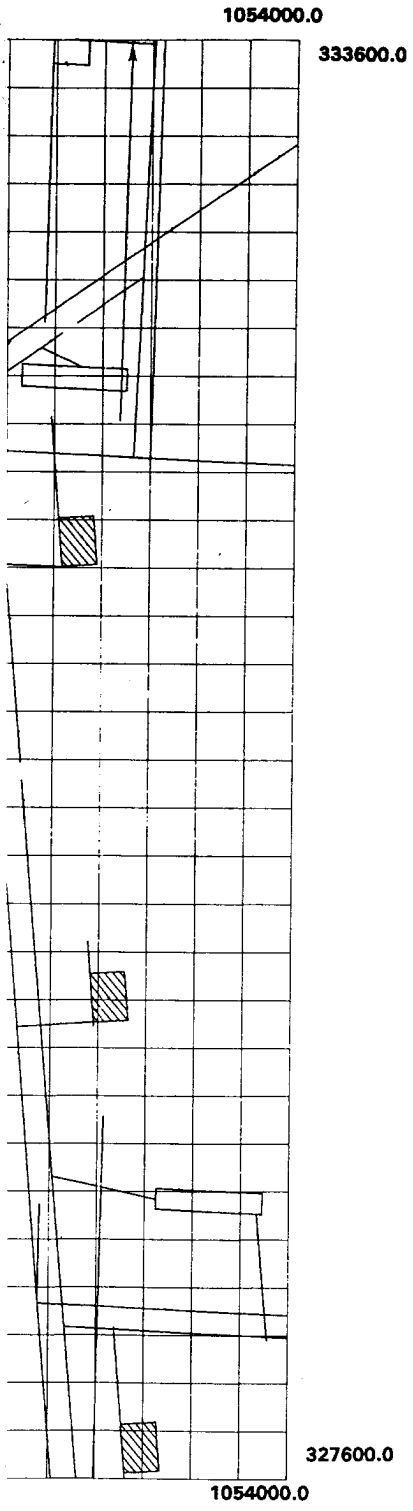
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Holiday Areas
- Trench Areas
- Digital Imagery
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1048000.0
333600.0
327600.0
1048000.0



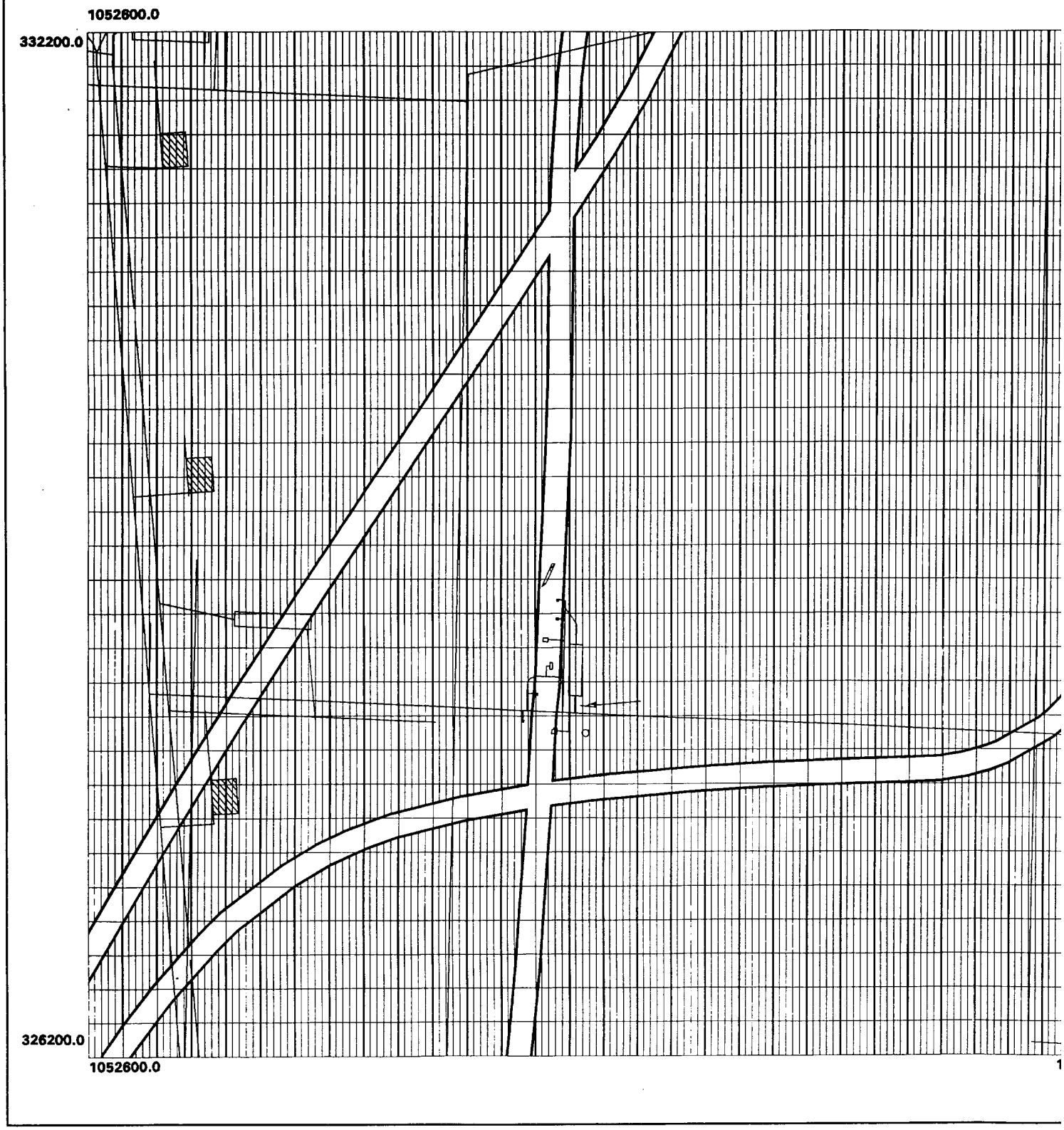
South Burning Ground 3
3 Satellites Available
Test Flight 3 &
Test Flight 4

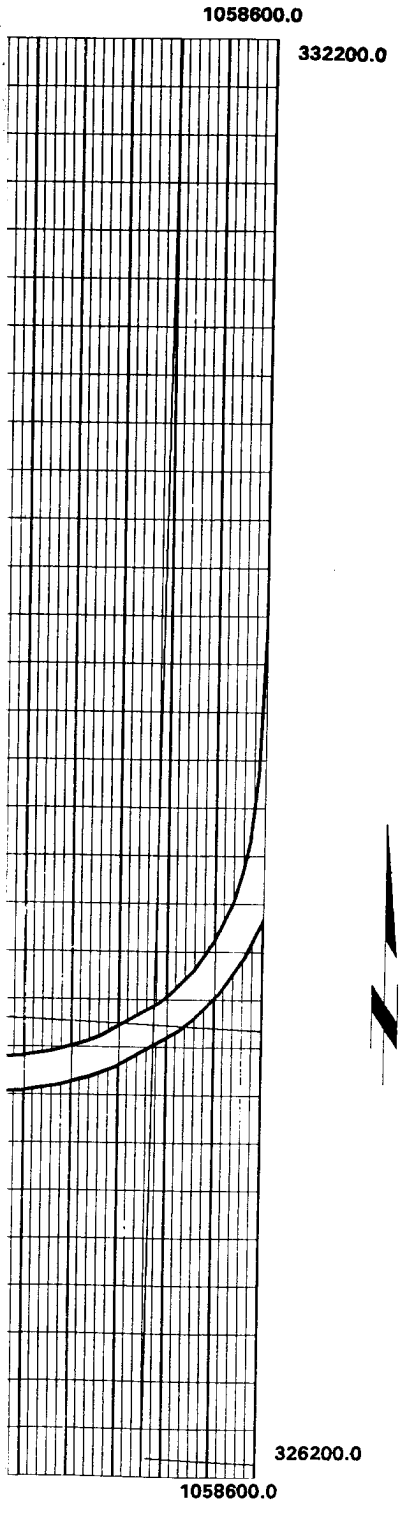
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- **Single Targets**
- **Light Density Target Areas**
- **Trench Areas**
- **Surface Anomalies**
- **Scattered Debris**
- **High Density Target Areas**


1052600.0
332200.0
326200.0
1052600.0



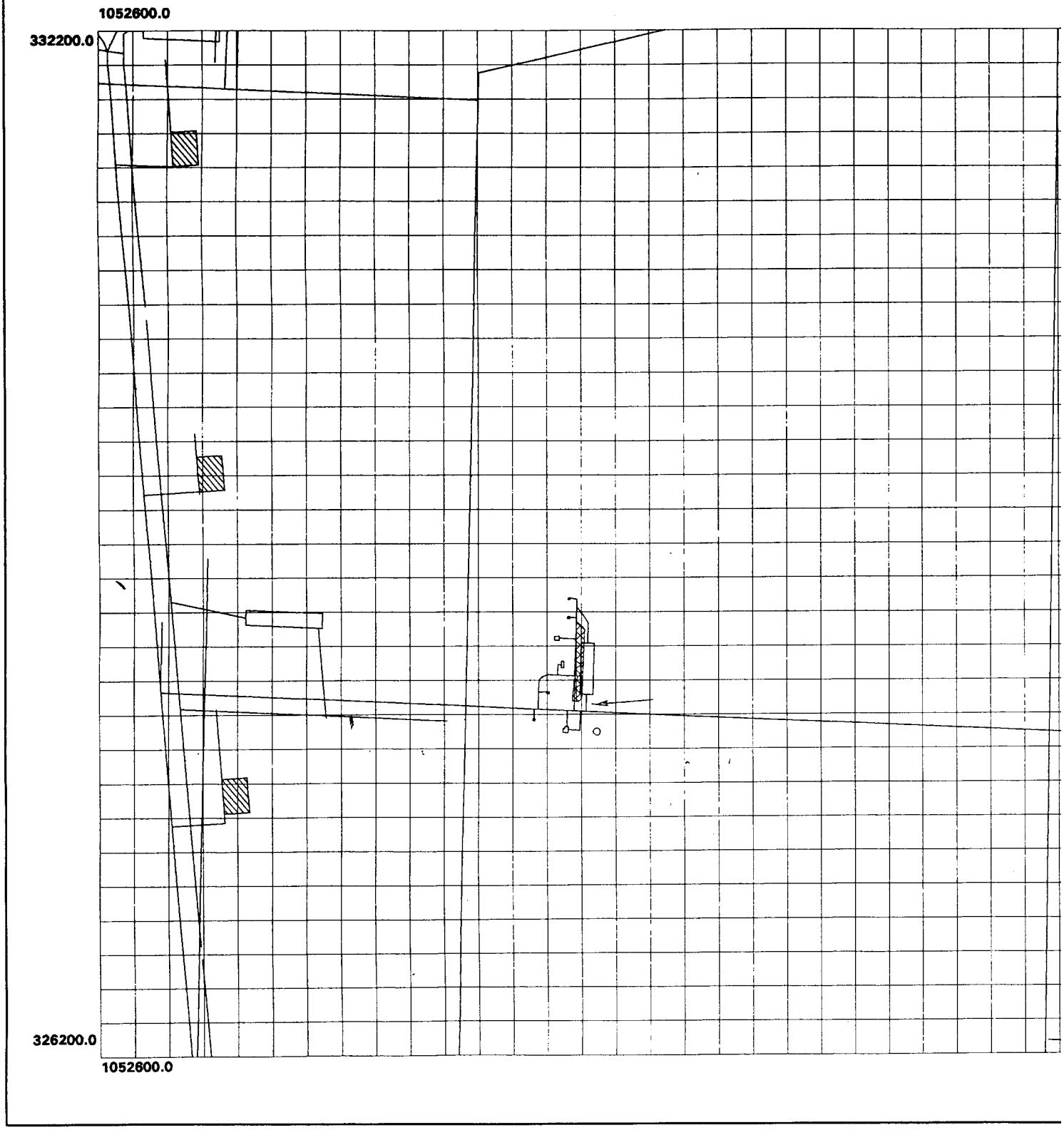
M06 Disassembly Plant Area
4 Satellites Available
Test Flight 6

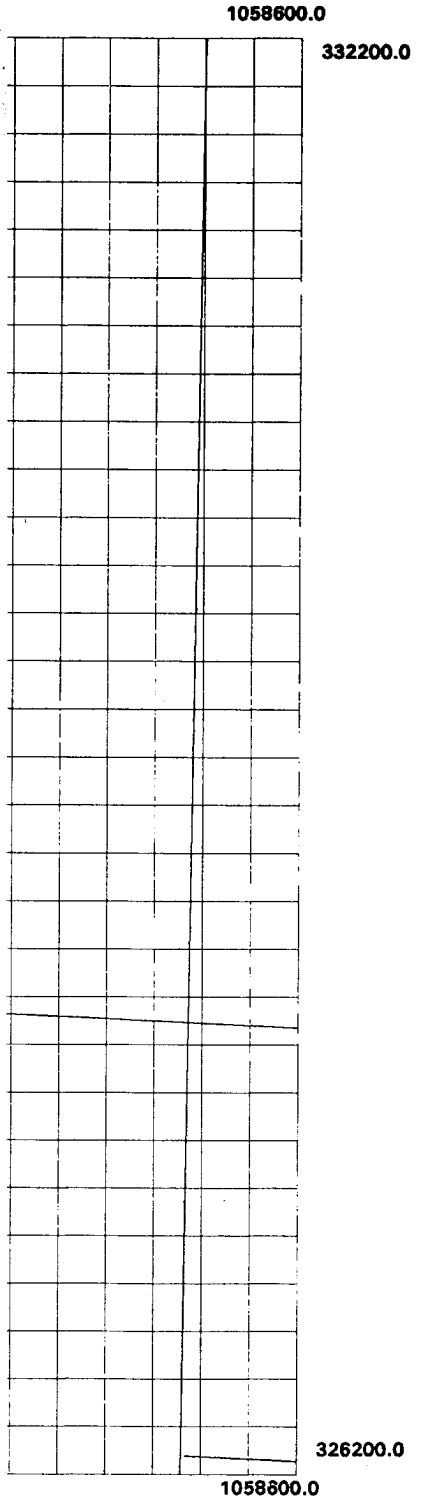
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery


1052600.0
332200.0
326200.0
1052600.0



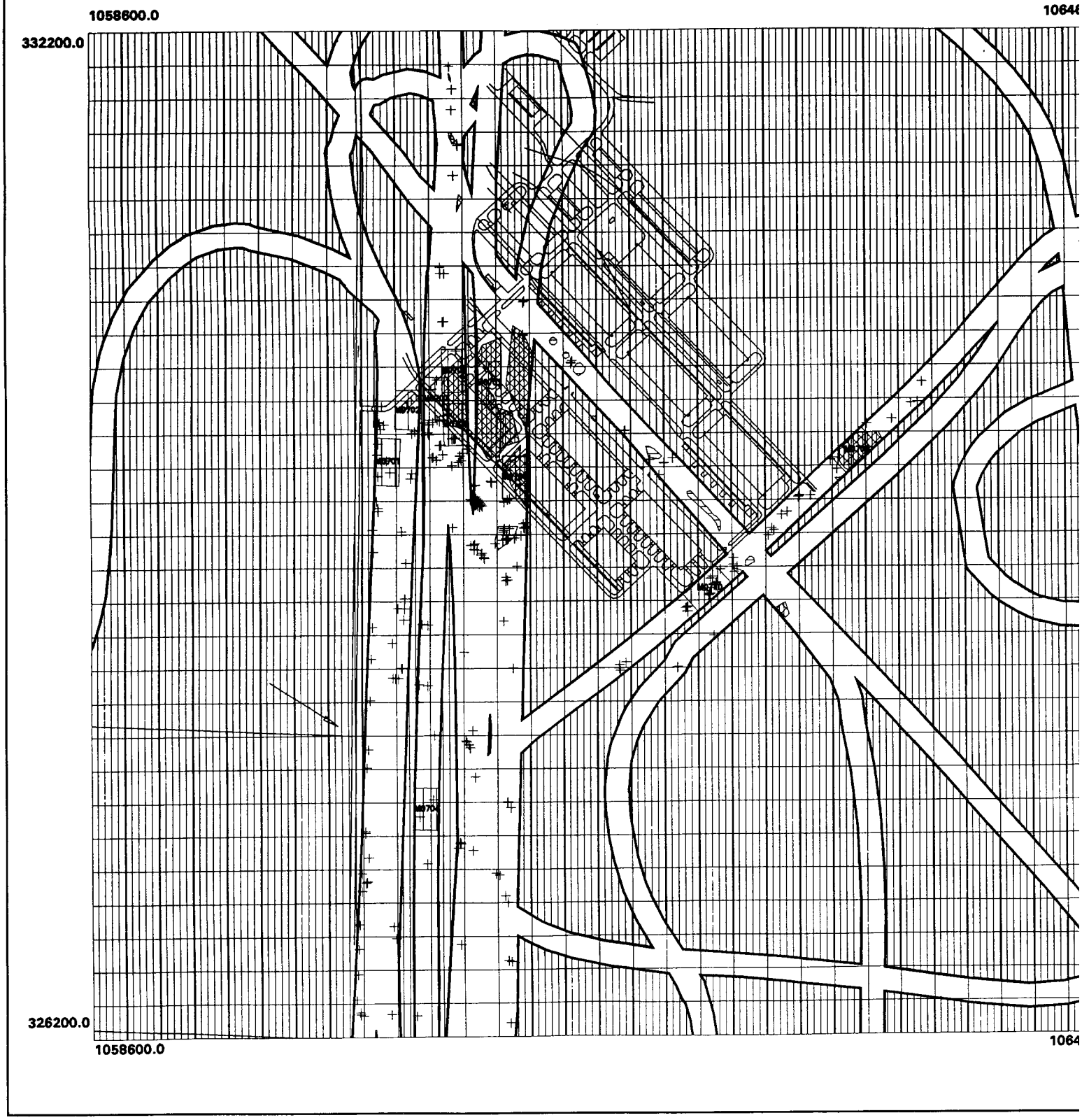
Disassembly Plant Area
3 Satellites Available
Test Flight 6

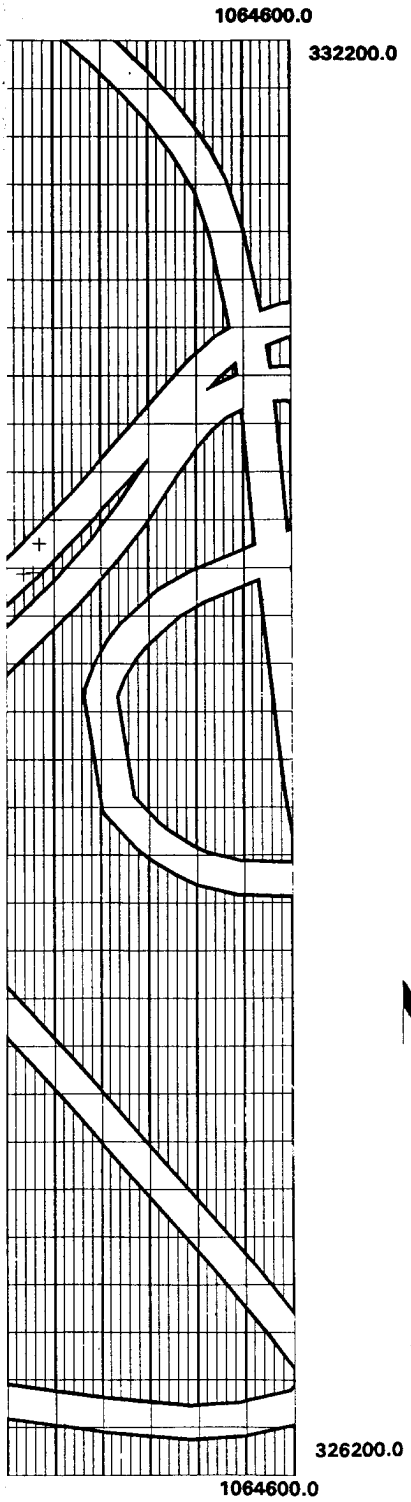
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

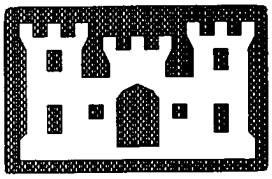
- **Single Targets**
- **Light Density Target Areas**
- **Trench Areas**
- **Surface Anomalies**
- **Scattered Debris**
- **High Density Target Areas**

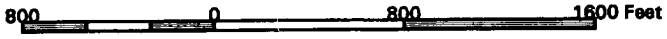

1058600.0
332200.0
326200.0
1058600.0



M07 Building 8000 Area
4 Satellites Available
Test Flight 6

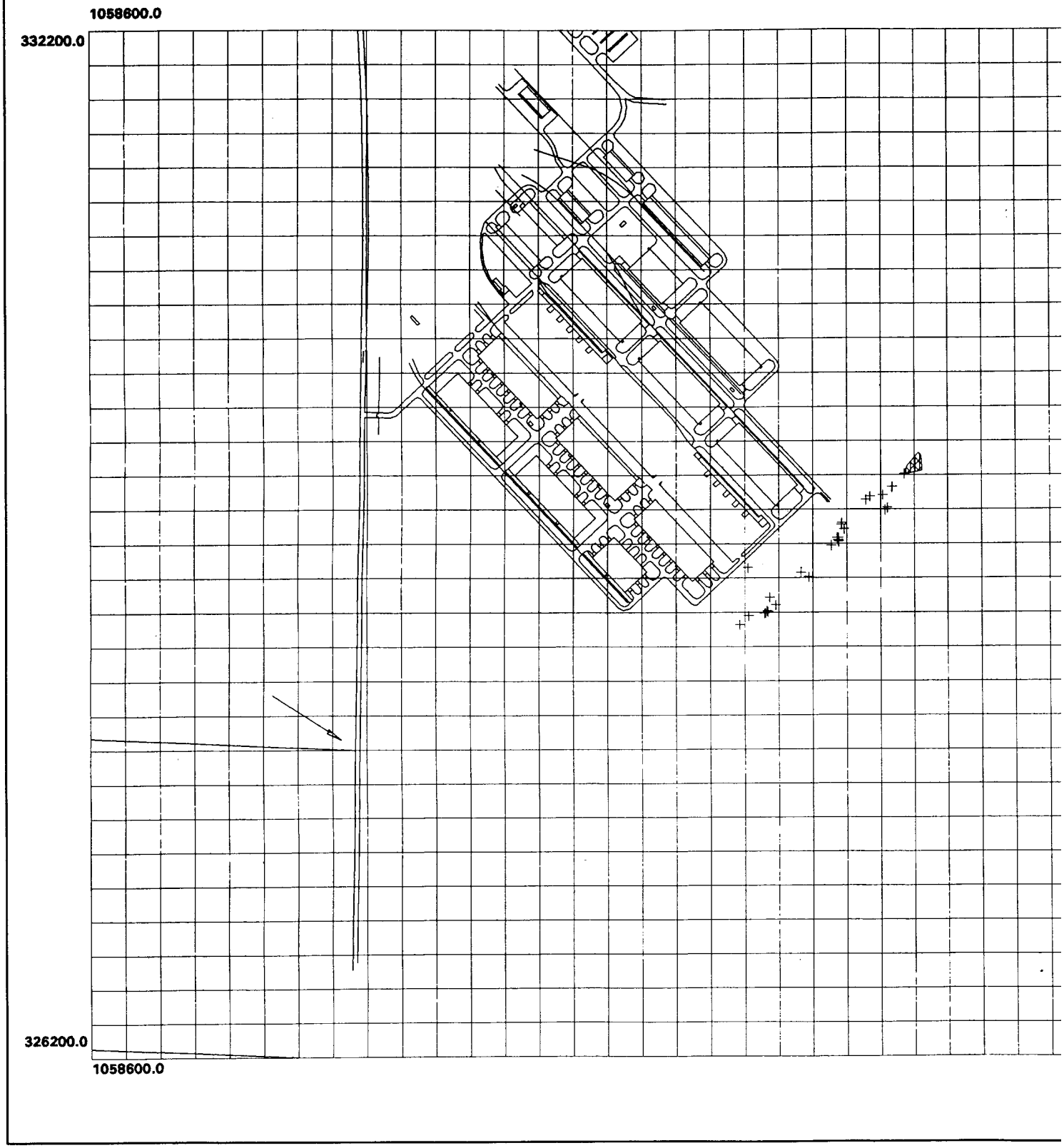
Release Date:
14 Oct 1994

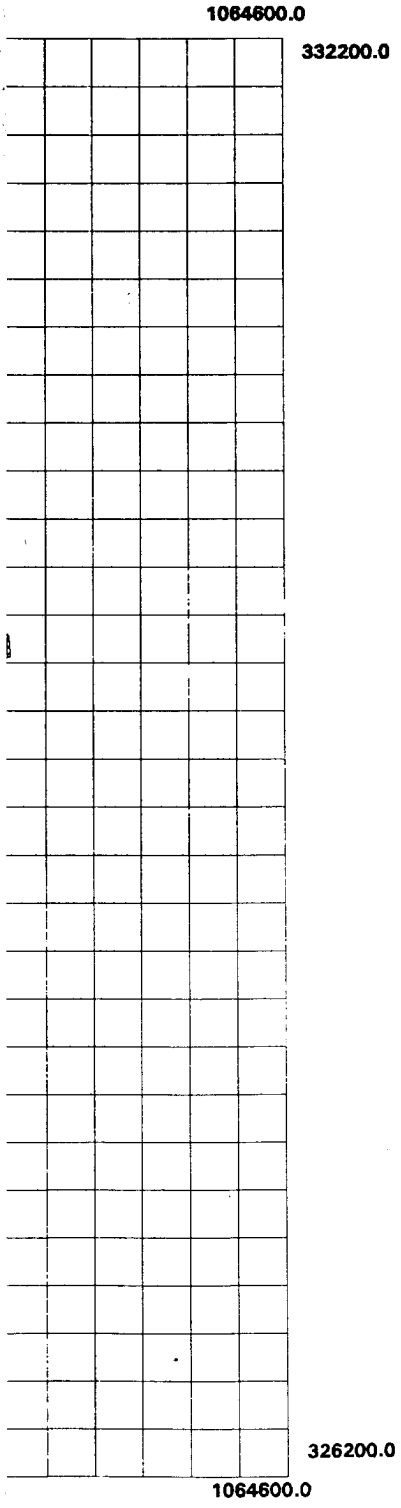




Legend

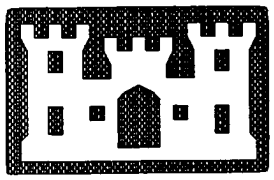
- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery

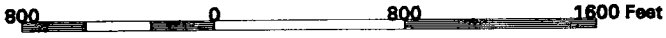

1058600.0
332200.0
326200.0
1058600.0



Building 8000 Area
3 Satellites Available
Test Flight 6

Release Date:
03 Dec 1993





Legend

Single Targets

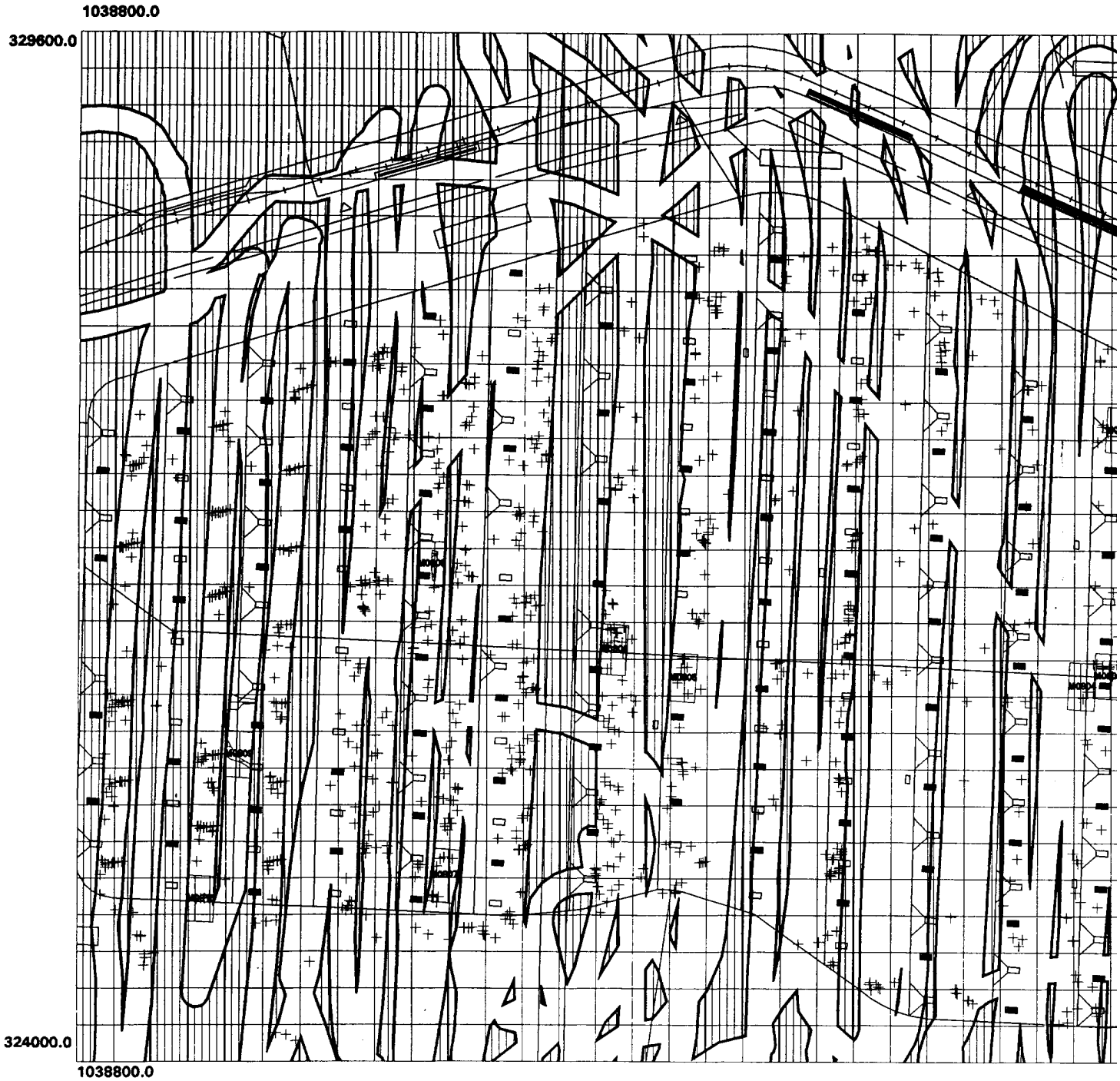
Light Density Target Areas

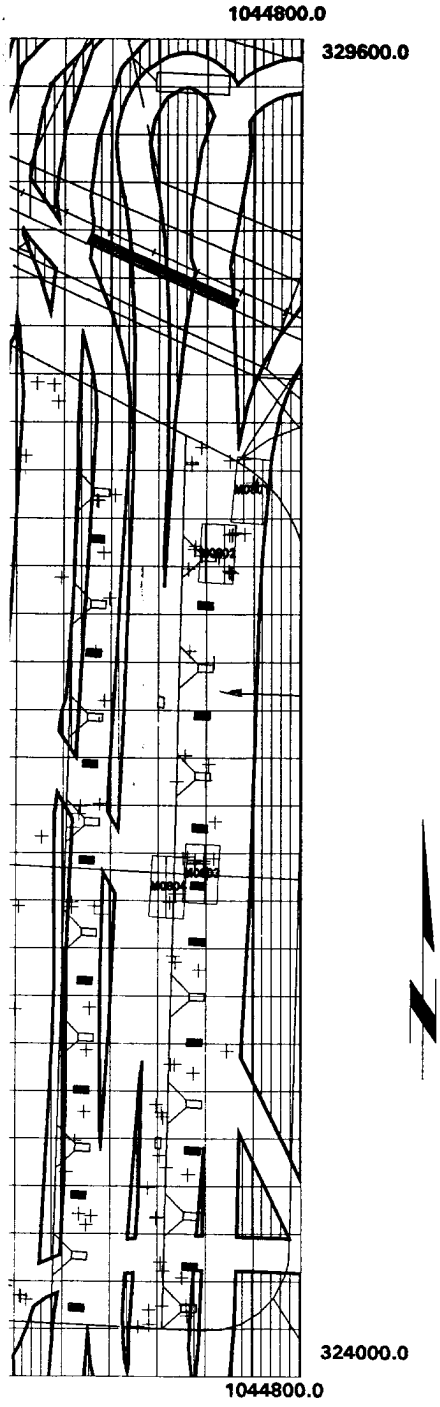
Trench Areas

Surface Anomalies

Scattered Debris

High Density Target Areas


1038800.0
329600.0
324000.0
1038800.0



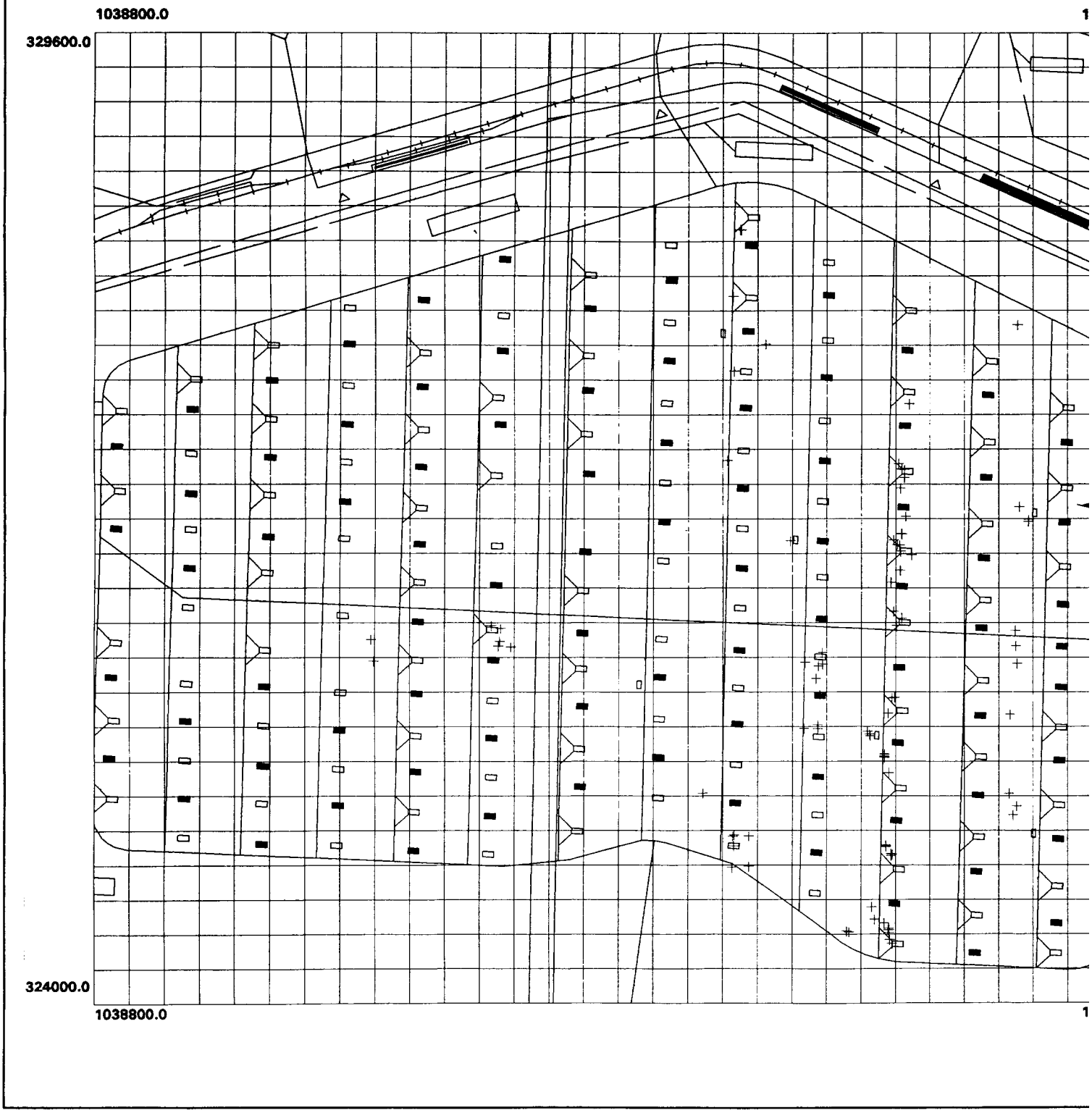
M08 Block F
4 Satellites Available
Test Flight 6

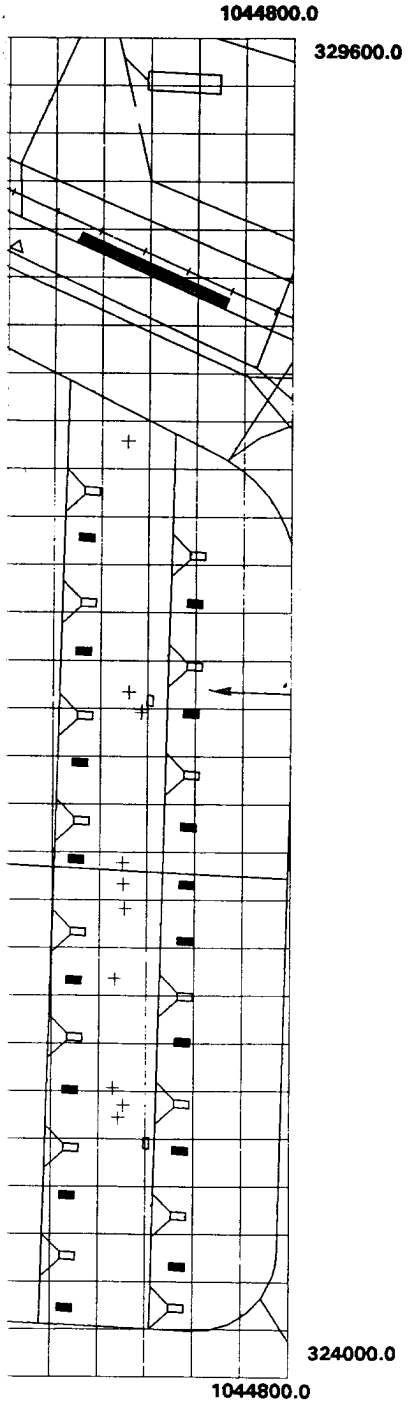
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Holiday Areas
- Trench Areas
- Digital Imagery
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1038800.0
329600.0
324000.0
1038800.0



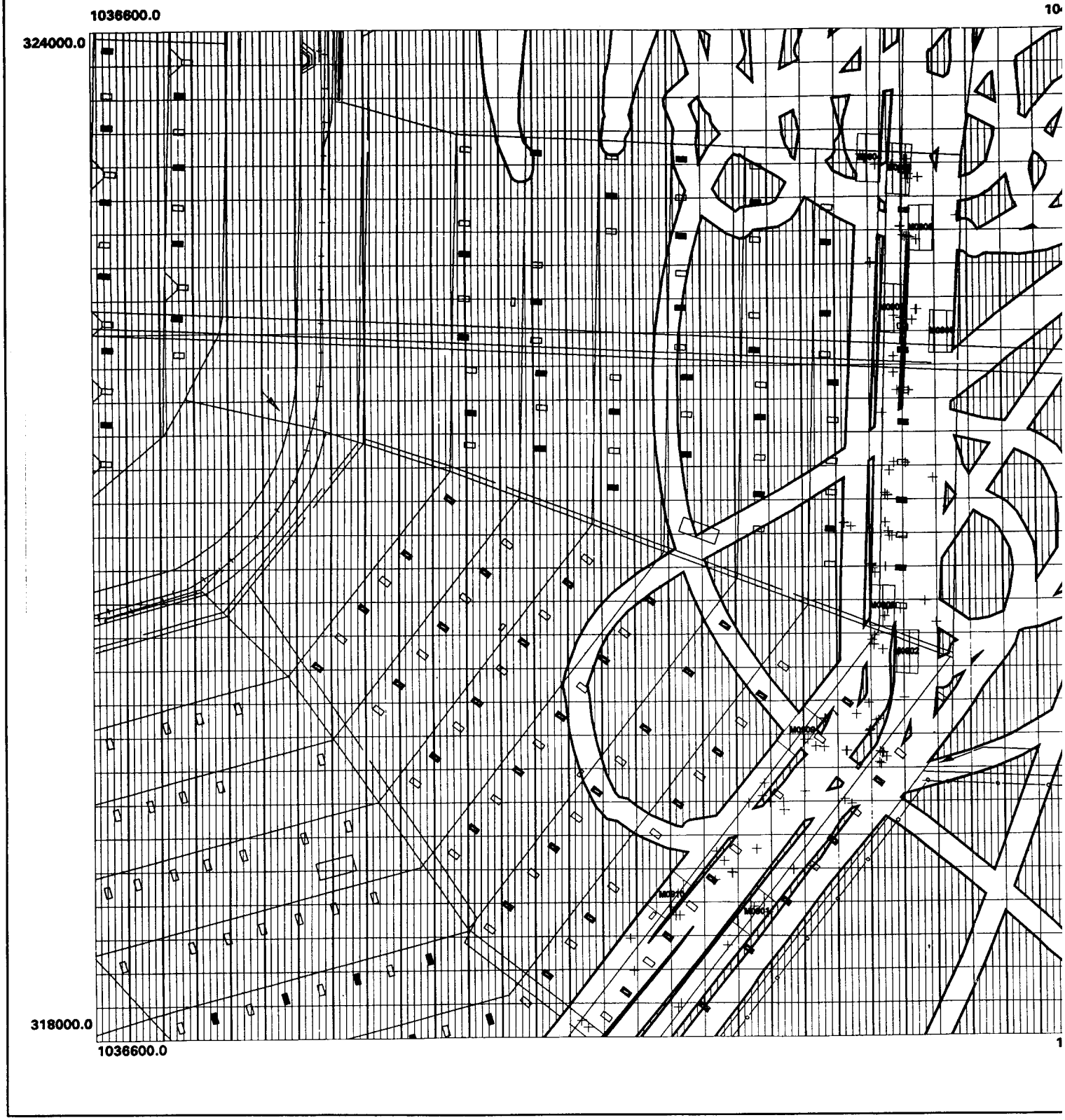
Block F
3 Satellites Available
Test Flight 6

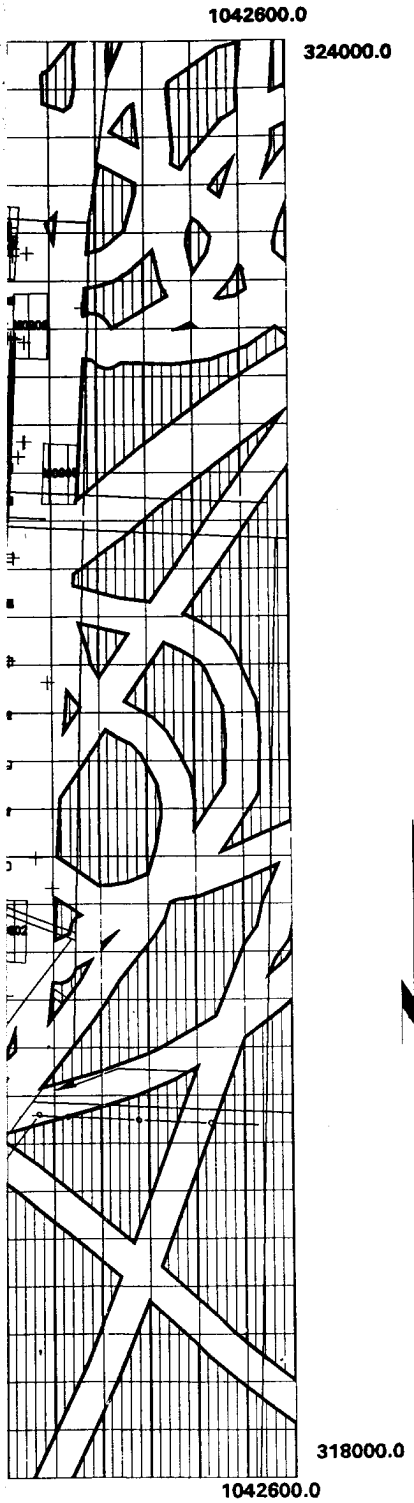
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1036600.0
324000.0
318000.0
1036600.0



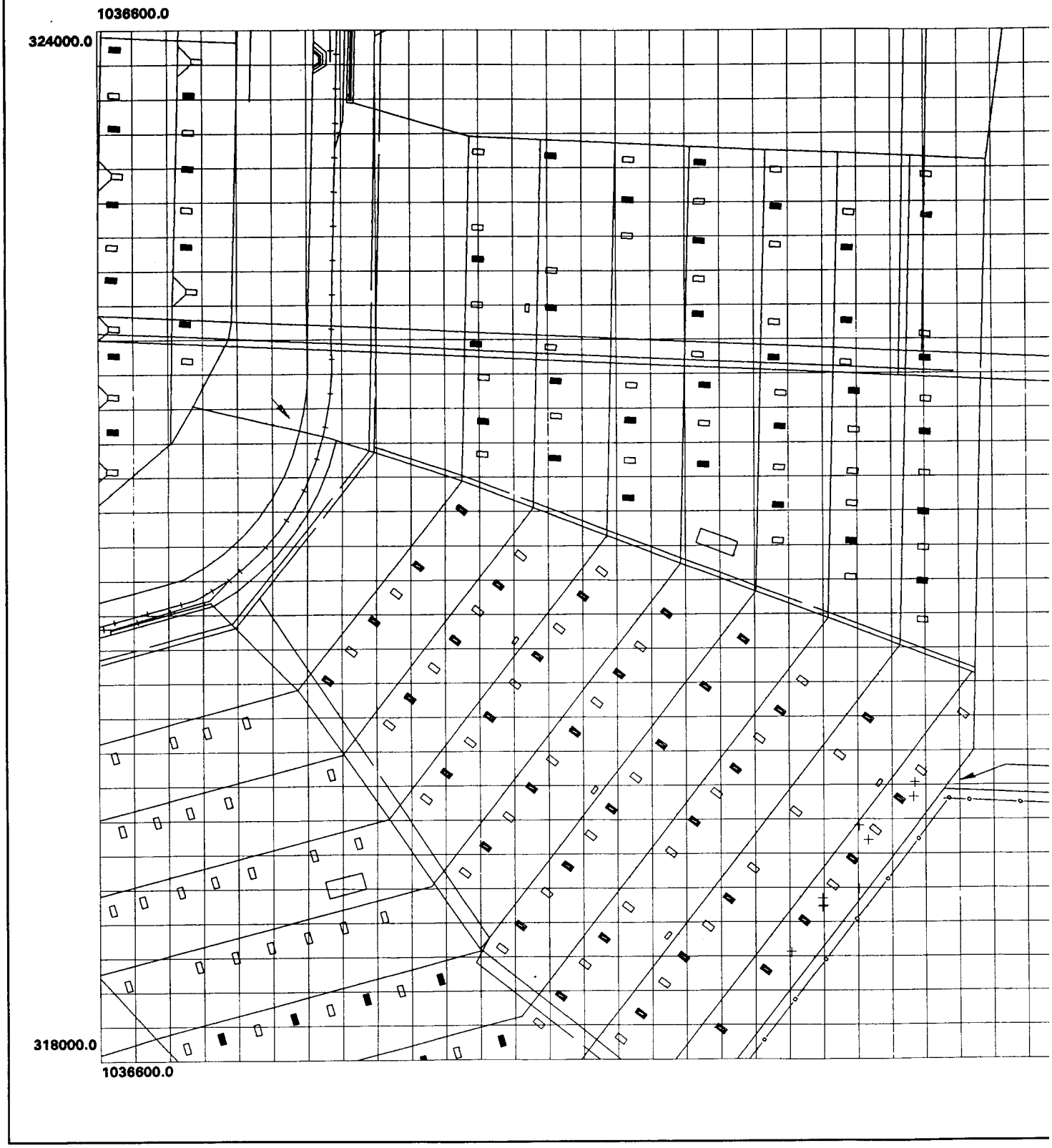
M09 Block D
4 Satellites Available
Test Flight 6

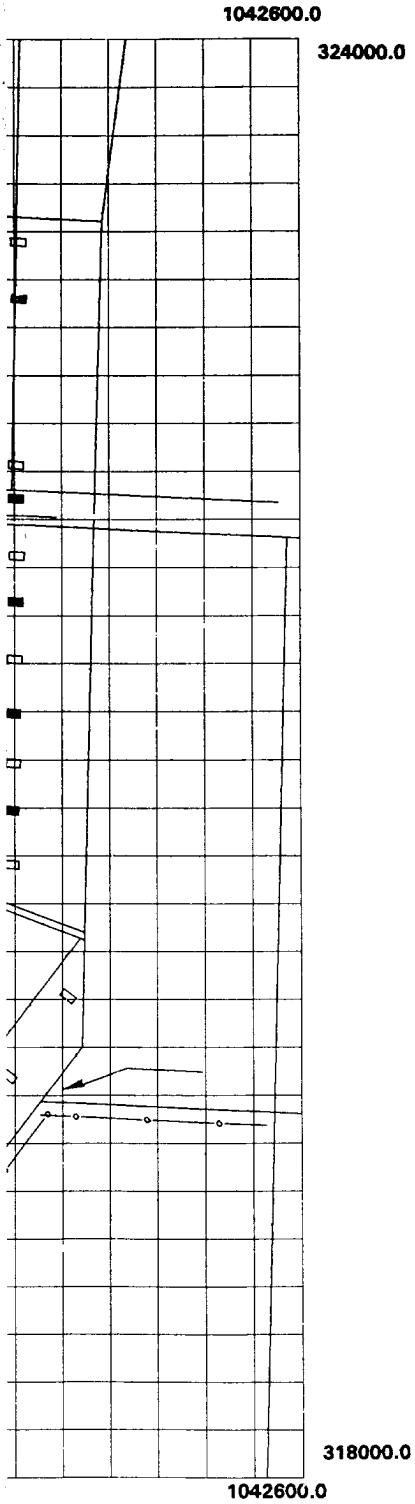
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery


1036600.0
324000.0
318000.0
1036600.0



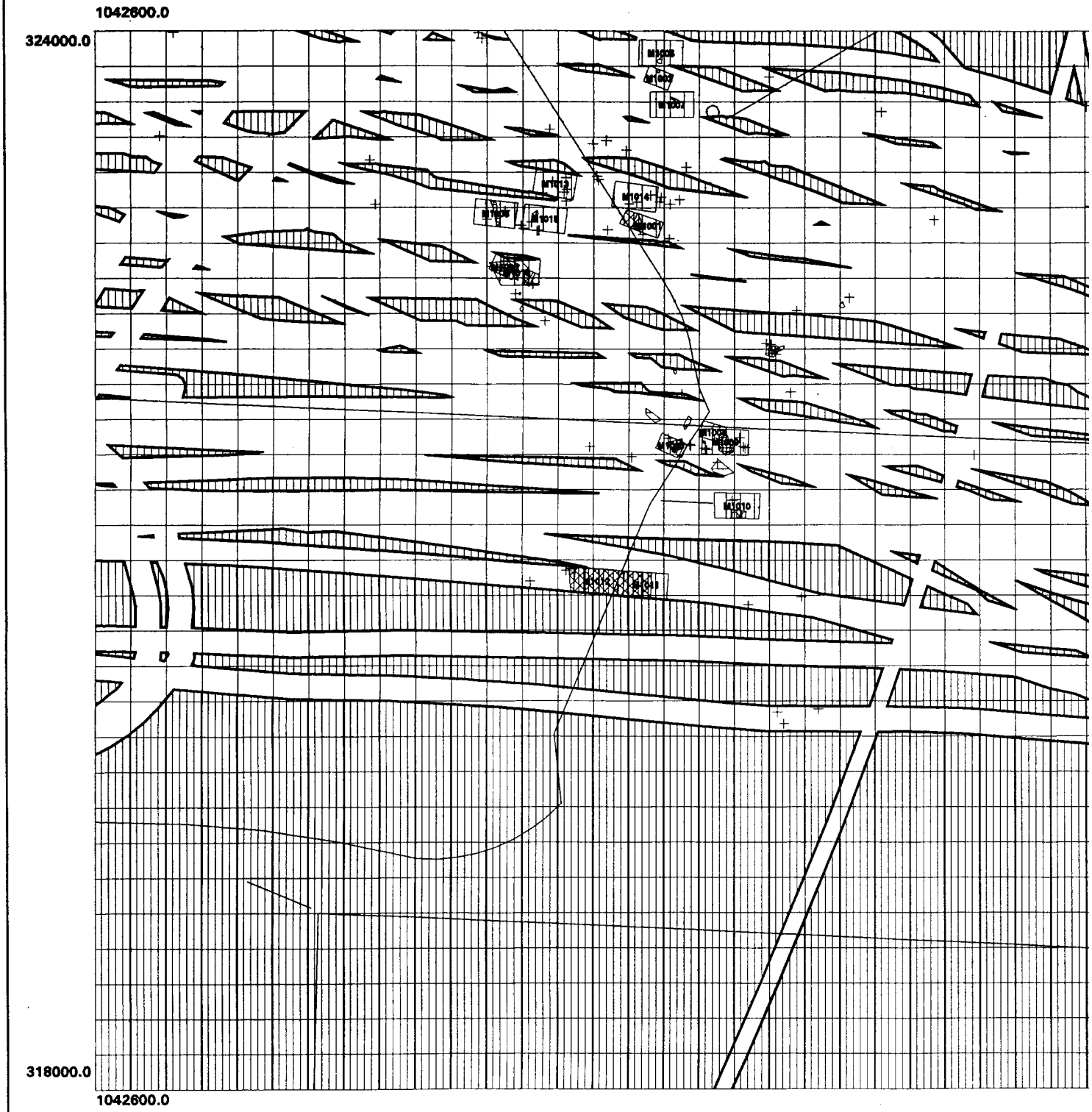
Block D
3 Satellites Available
Test Flight 6

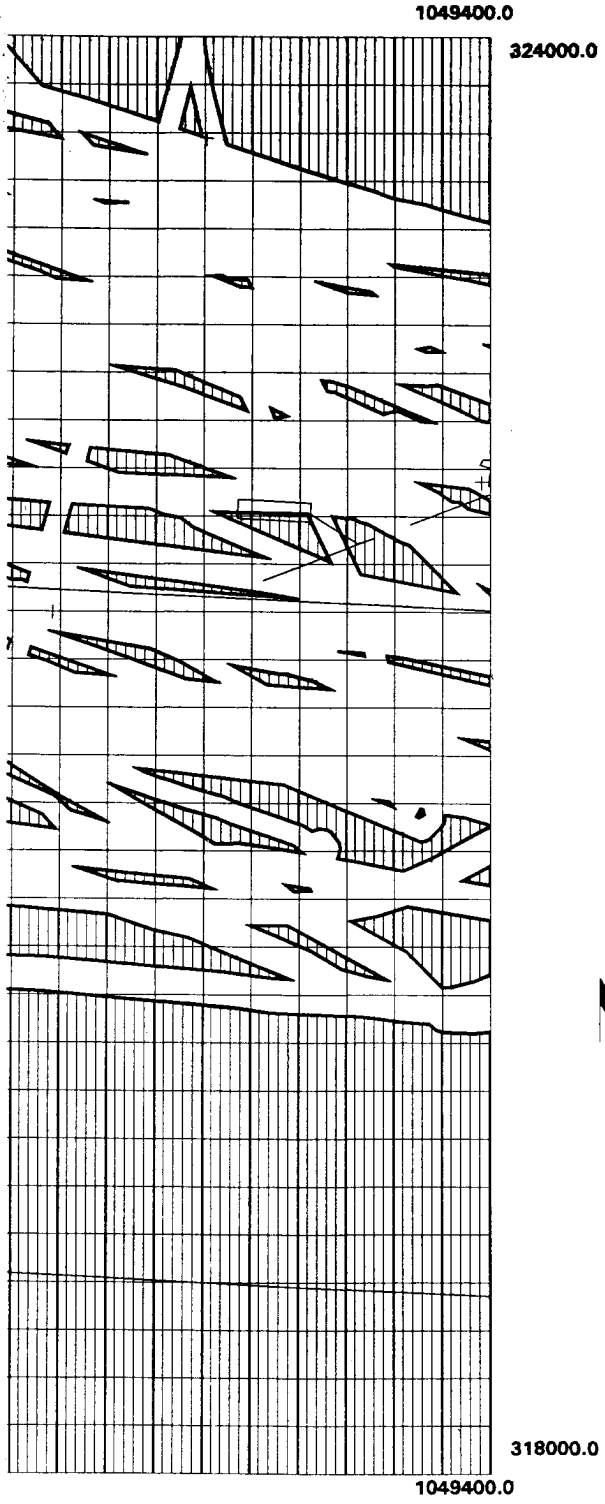
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- **Single Targets**
- **Light Density Target Areas**
- **Trench Areas**
- **Surface Anomalies**
- **Scattered Debris**
- **High Density Target Areas**


1042600.0
324000.0
318000.0
1042600.0



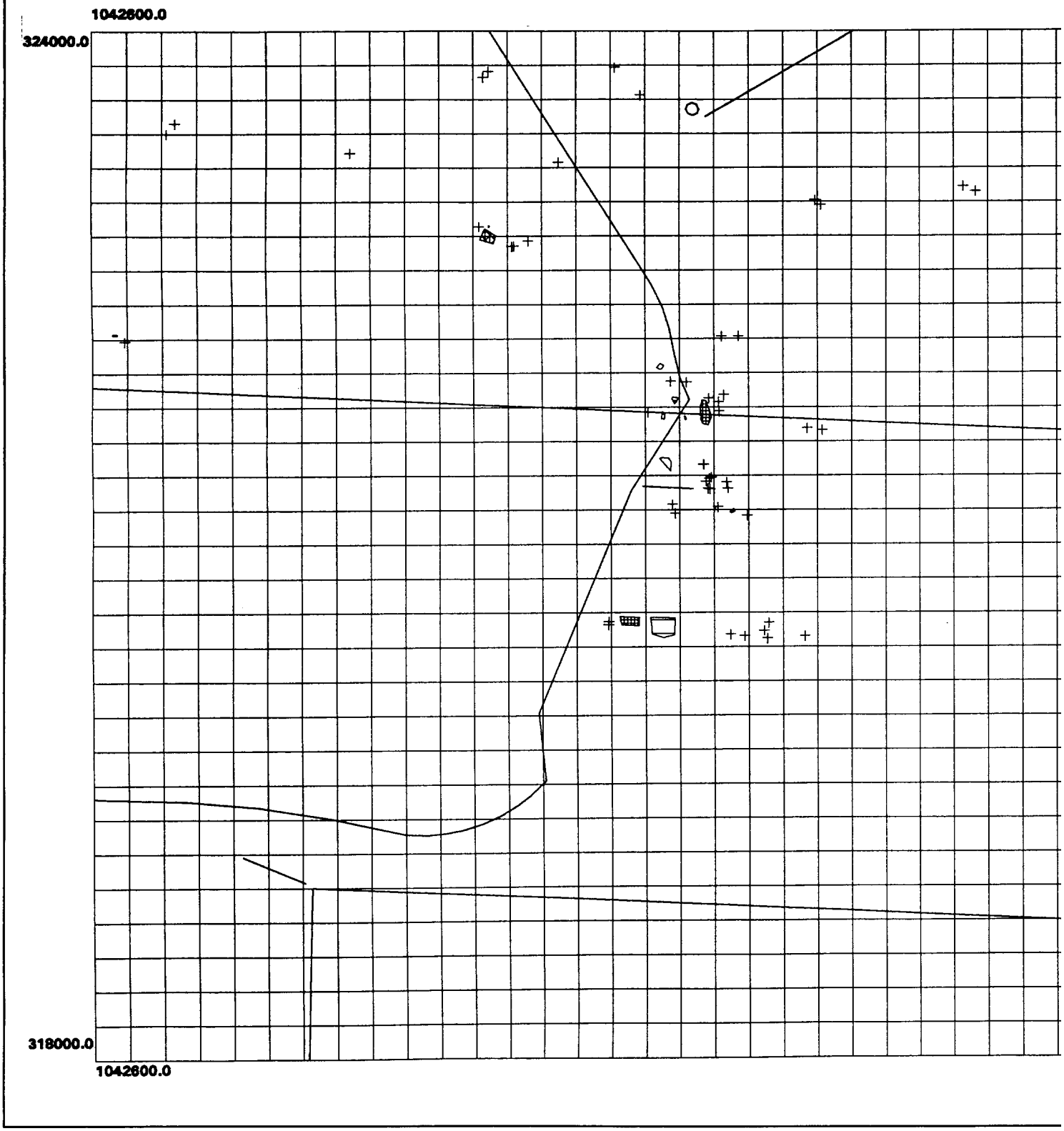
M10 Burning Ground 1
4 Satellites Available
Test Flight 4 &
Test Flight 5

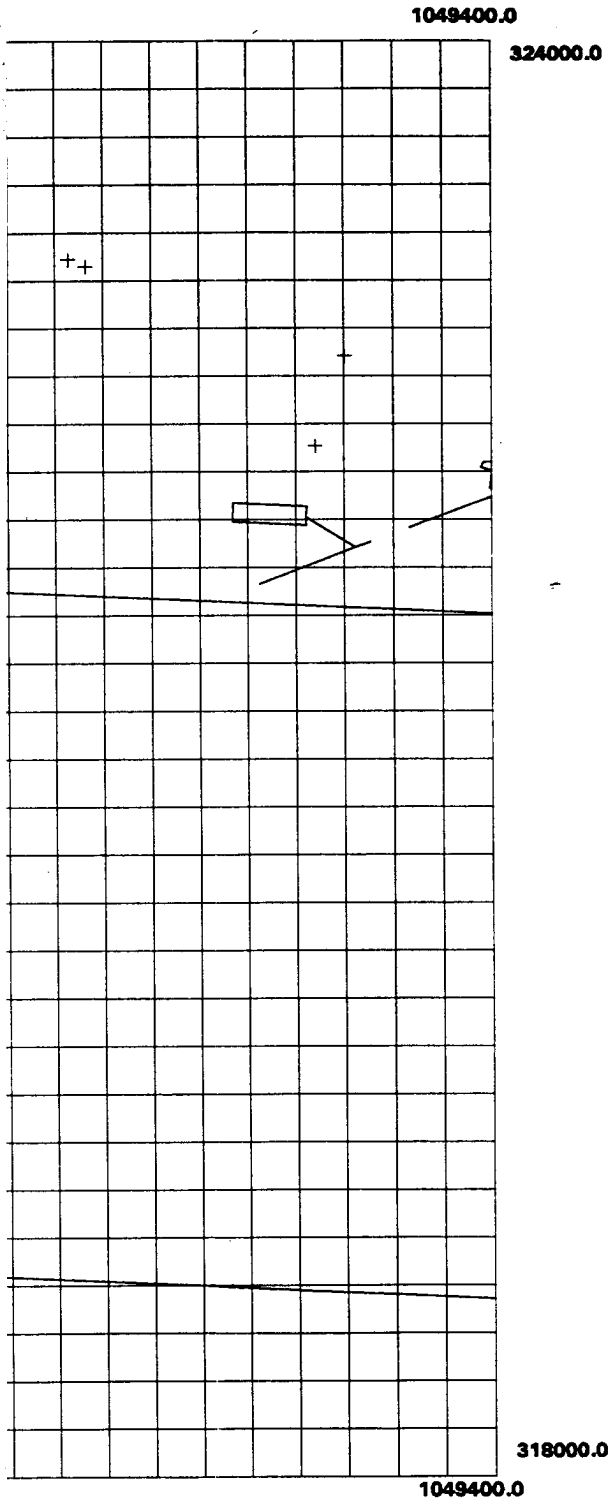
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery


1042600.0
324000.0
318000.0
1042600.0



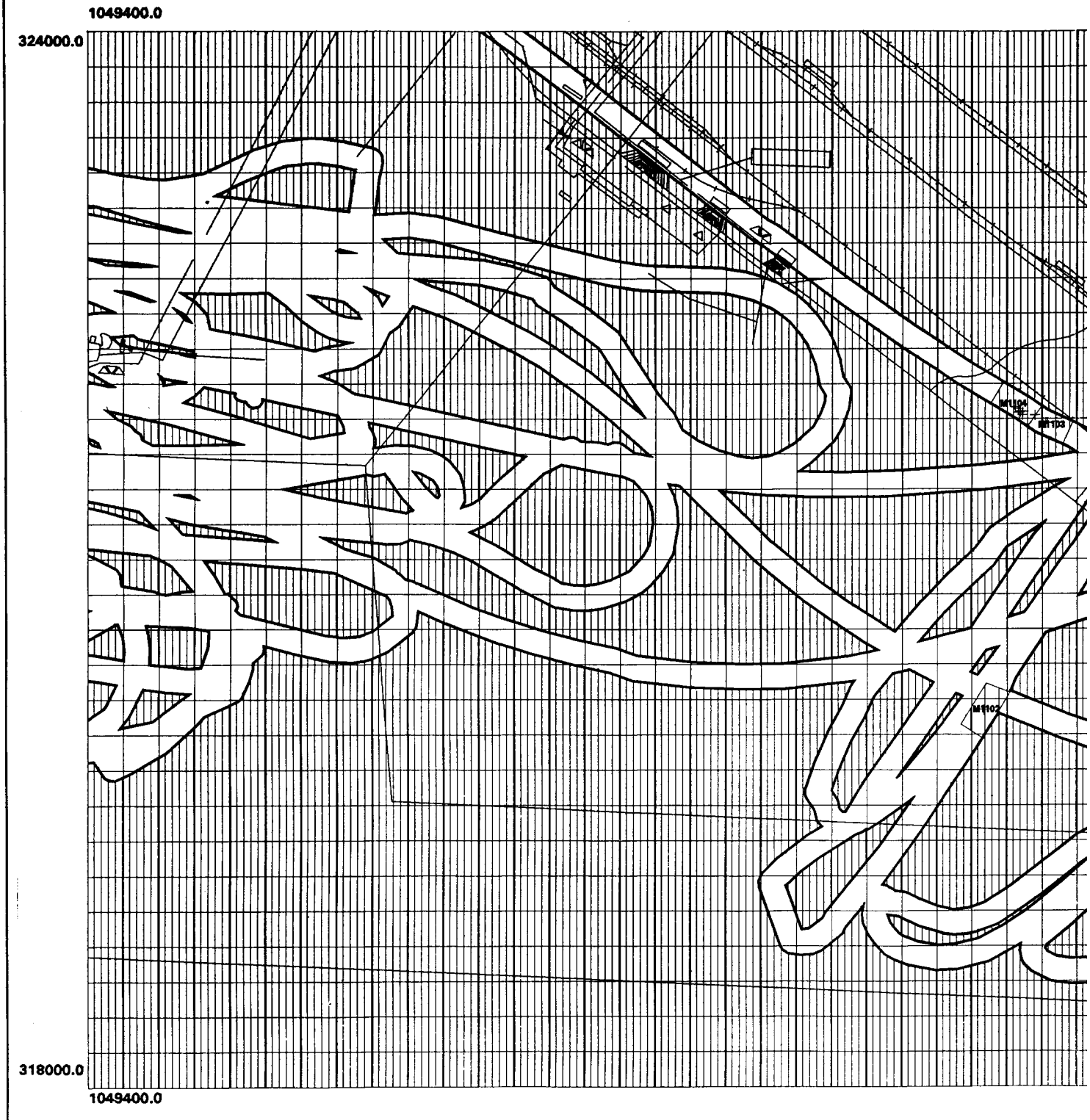
Burning Ground 1
3 Satellites Available
Test Flight 4 &
Test Flight 5

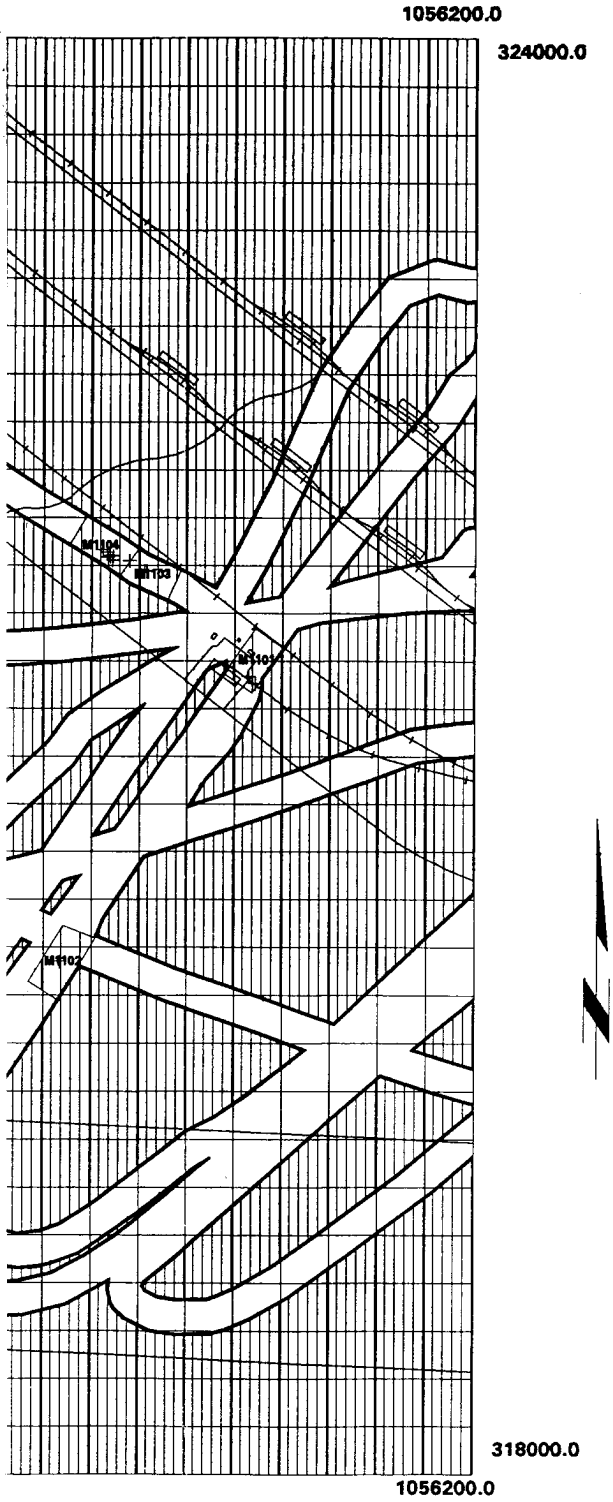
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1049400.0
324000.0
318000.0
1049400.0
M1104
M1103
M1102



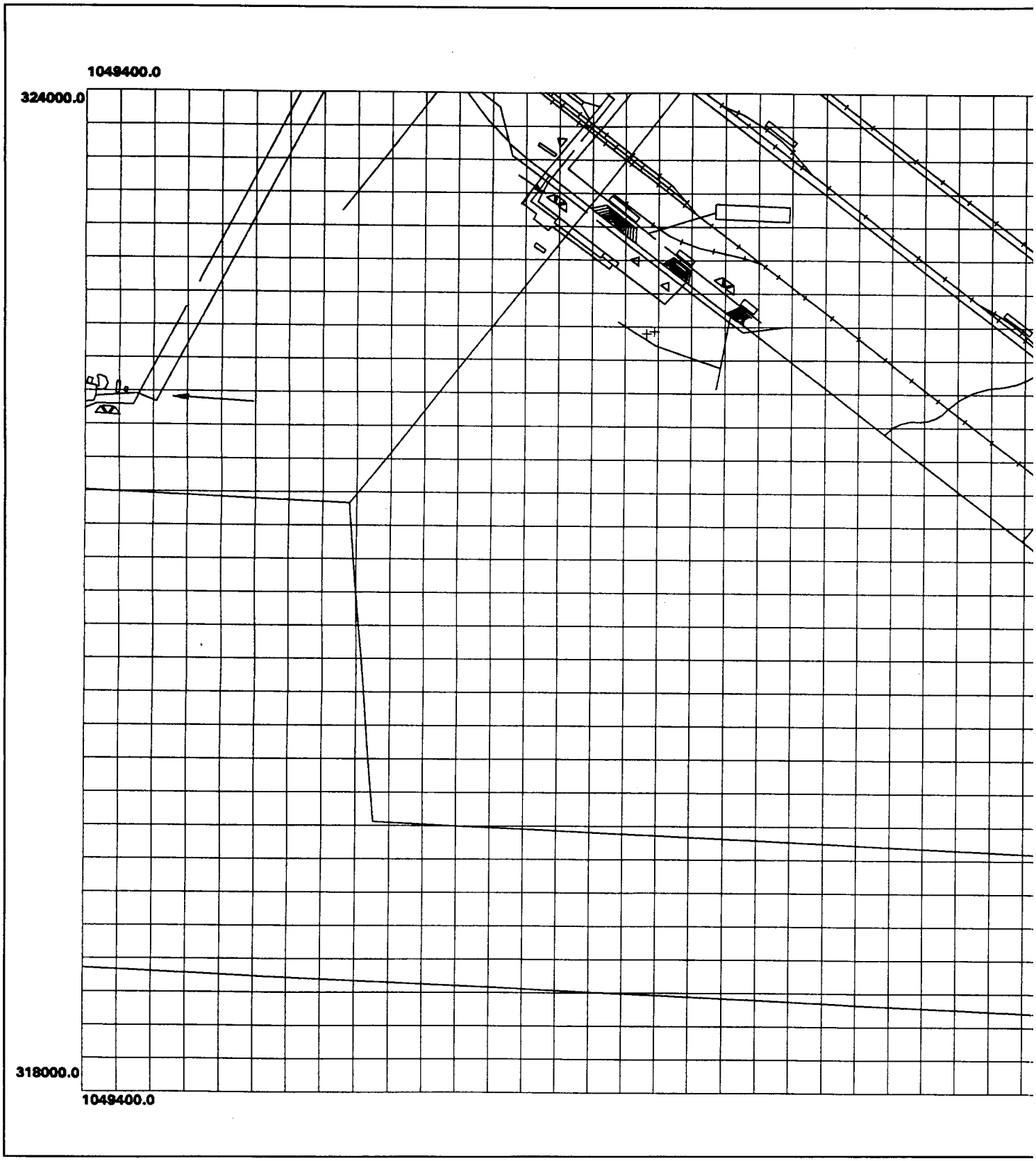
M11 East of Burning Ground 1
4 Satellites Available
Test Flight 4 &
Test Flight 5

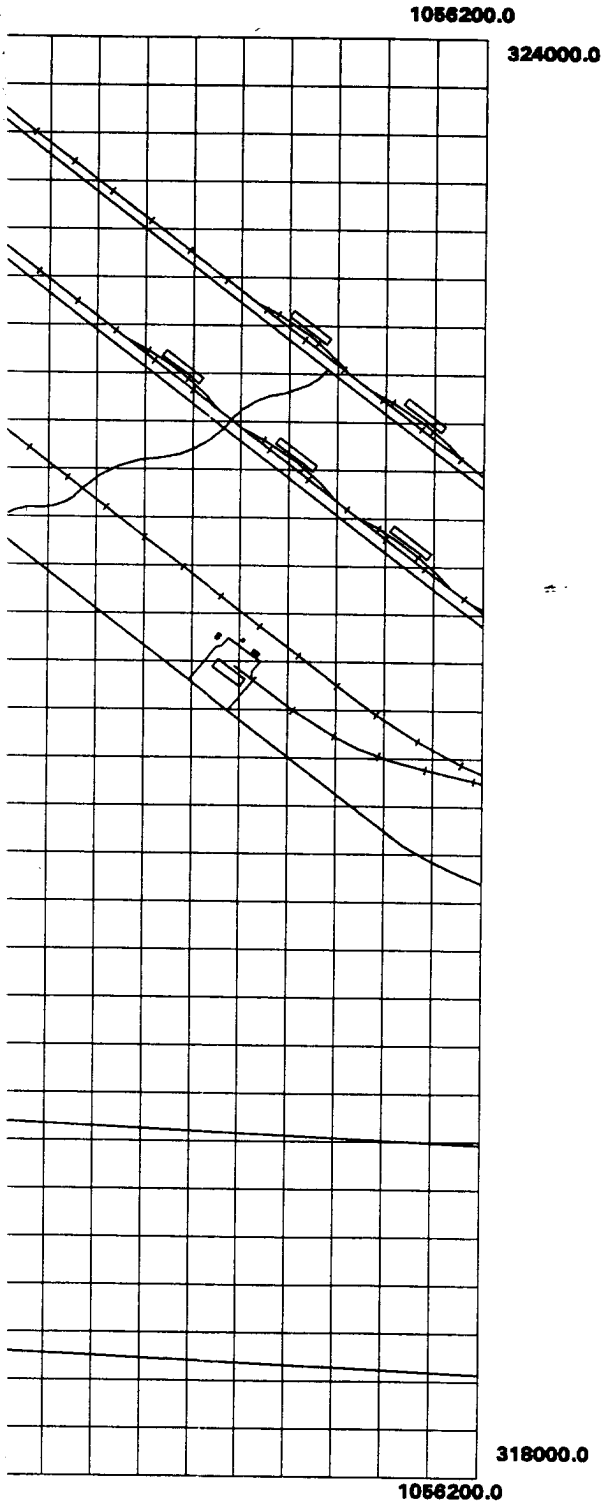
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery


1049400.0
324000.0
318000.0
1049400.0



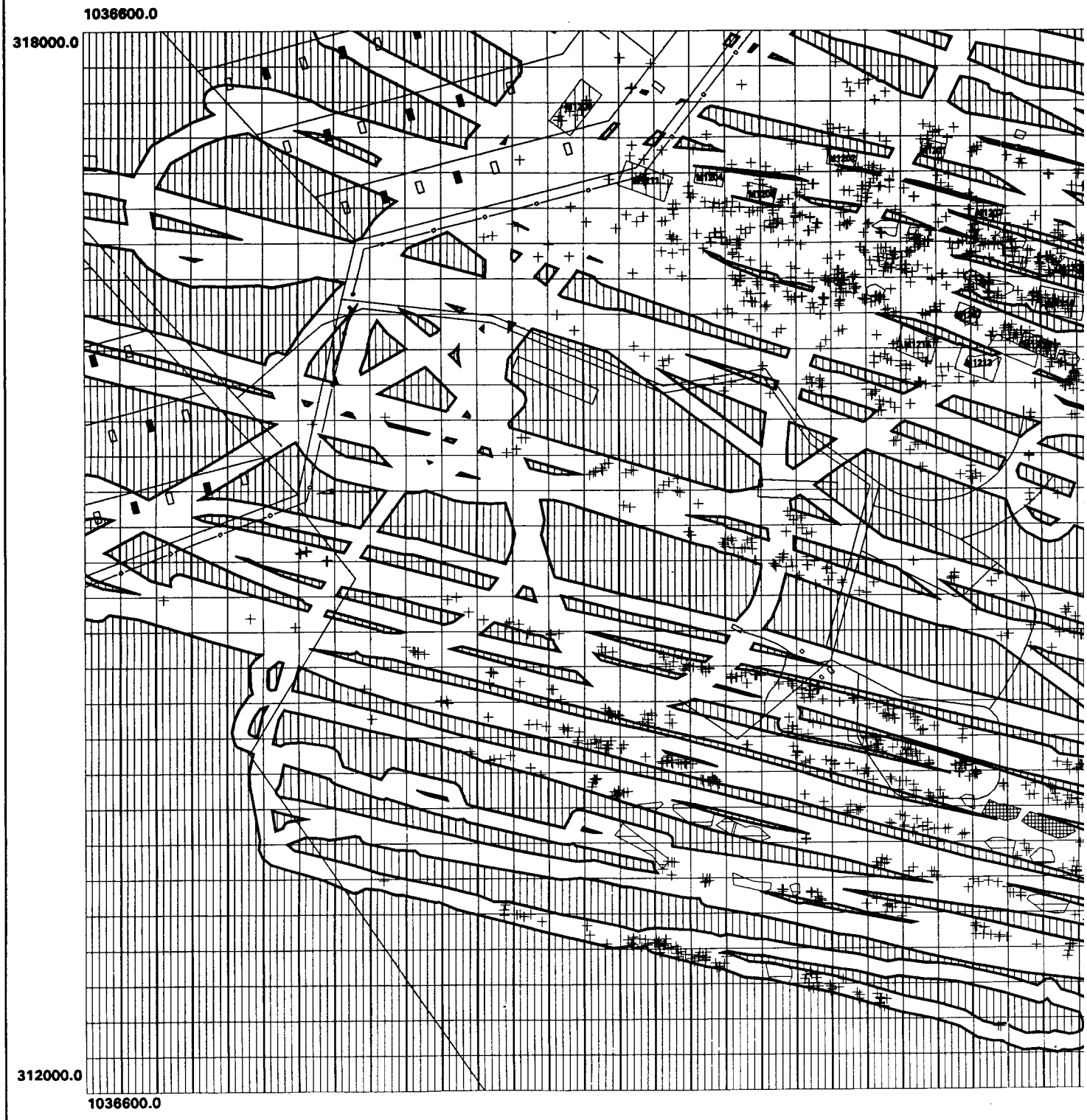
East of Burning Ground 1
3 Satellites Available
Test Flight 4 &
Test Flight 5

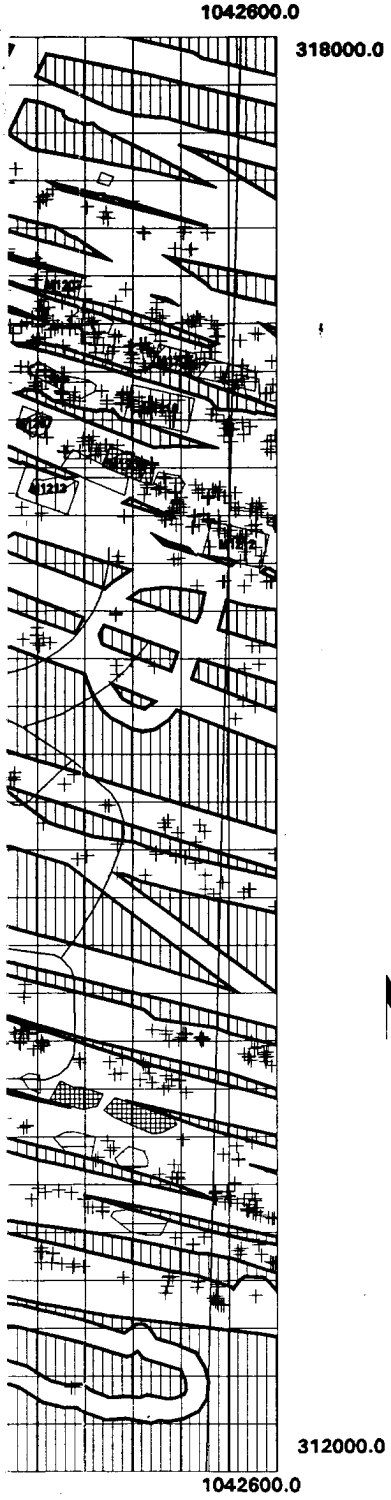
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1036600.0
318000.0
312000.0
1036600.0



M12 Burning Ground 2 & Block D

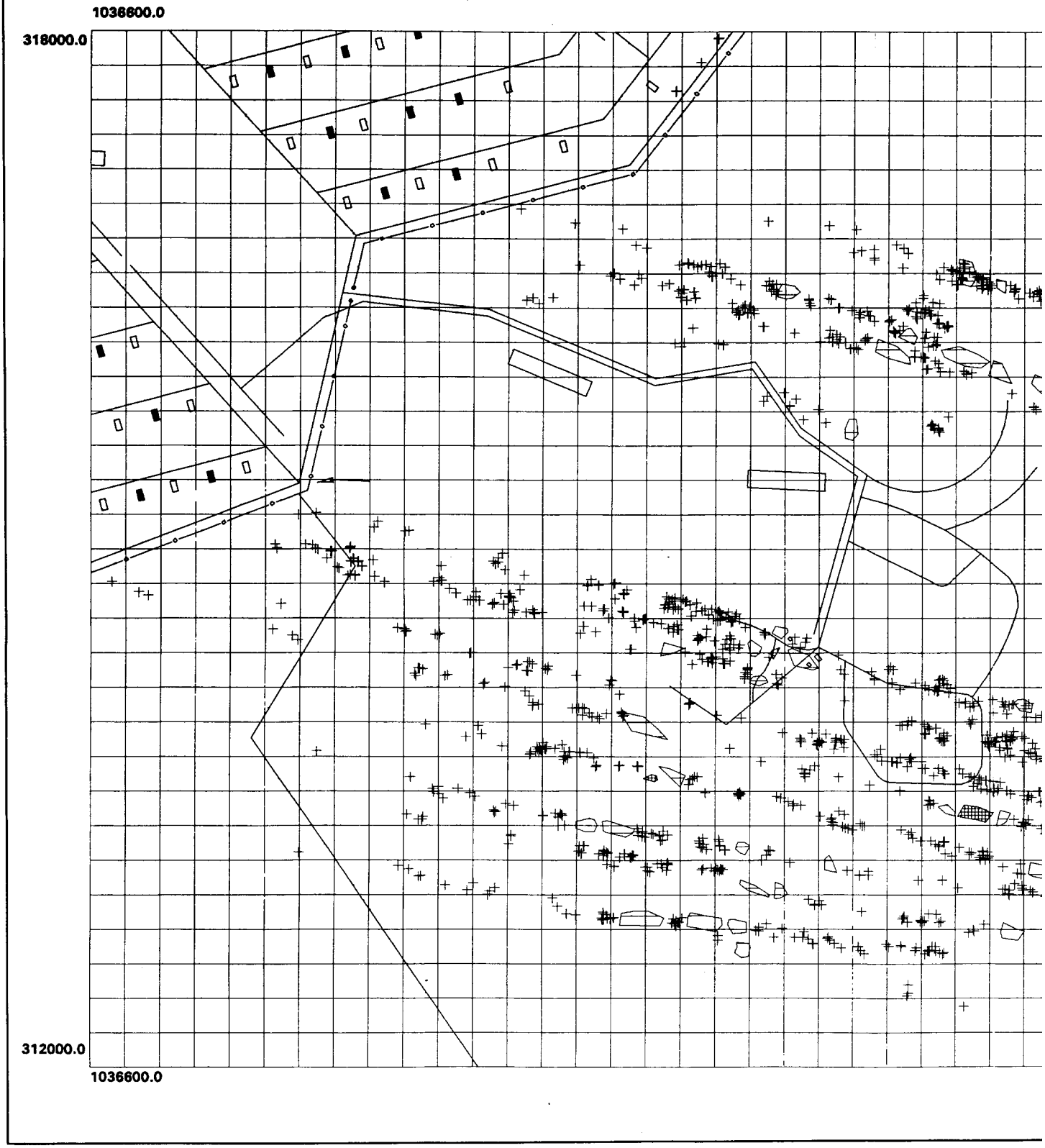
4 Satellites Available
Test Flight 1, Test Flight 2,
Test Flight 4, Test Flight 6, &
Test Flight 7

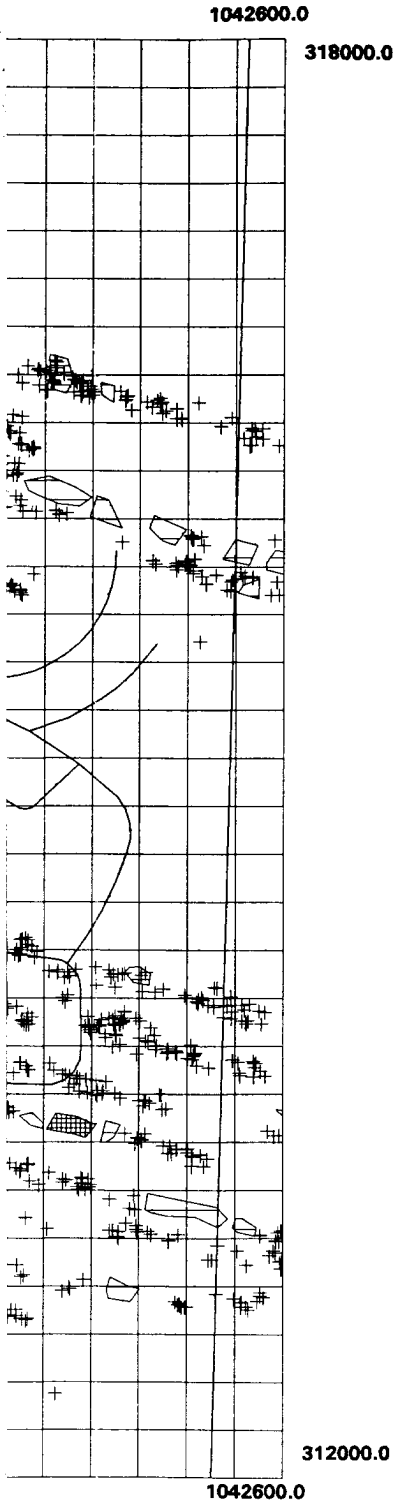
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Holiday Areas
- Trench Areas
- Digital Imagery
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1036600.0
318000.0
312000.0
1036600.0



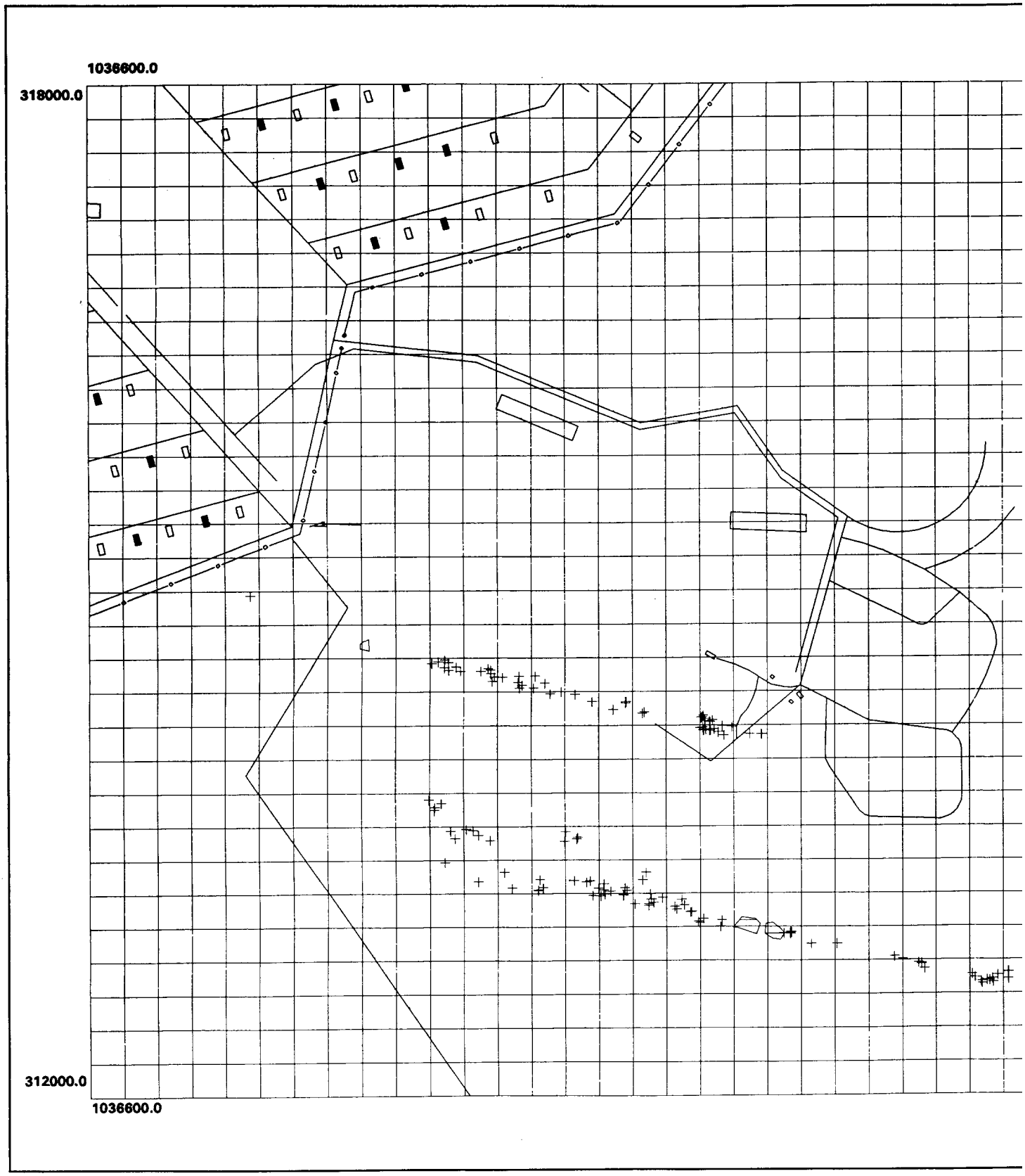
Burning Ground 2 & Block D
3 Satellites Available
Test Flight 1, Test Flight 2, Test Flight 4, Test Flight 6, & Test Flight 7

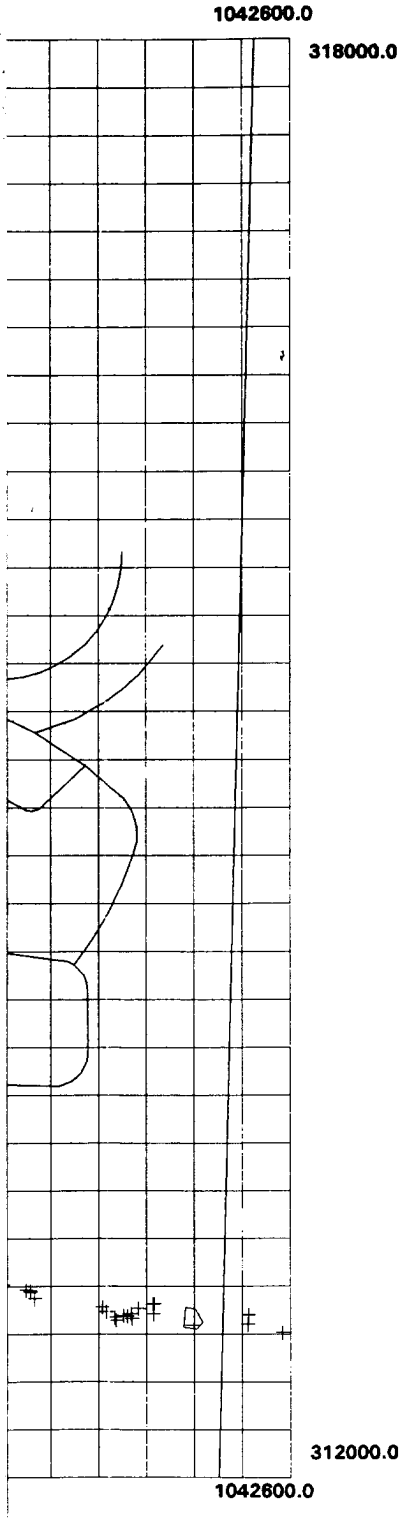
Release Date: 03 Dec 1993

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas


1036600.0
318000.0
312000.0
1036600.0



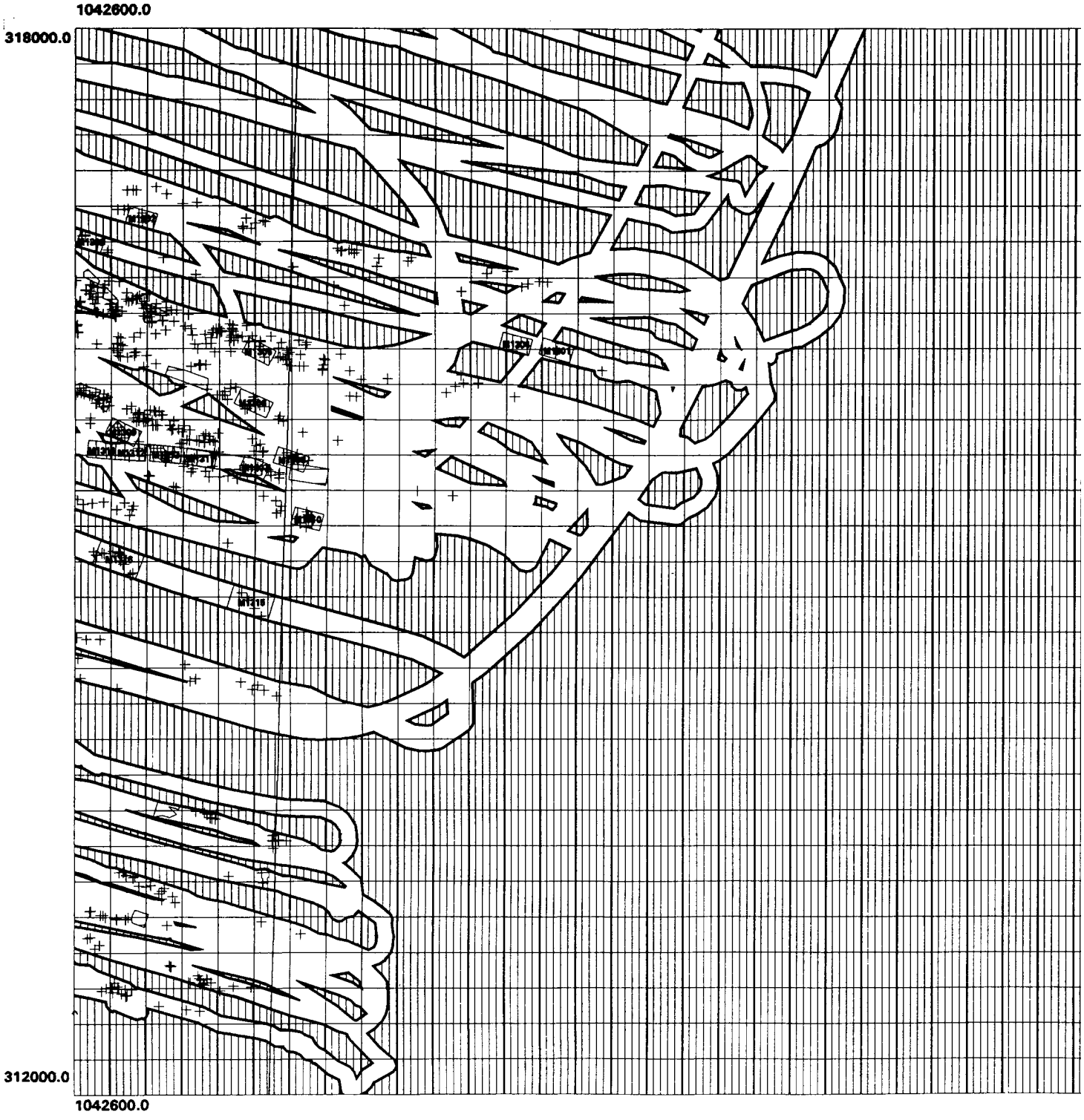
Burning Ground 2
Uncorrectable GPS Coverage
Test Flight 2 ONLY

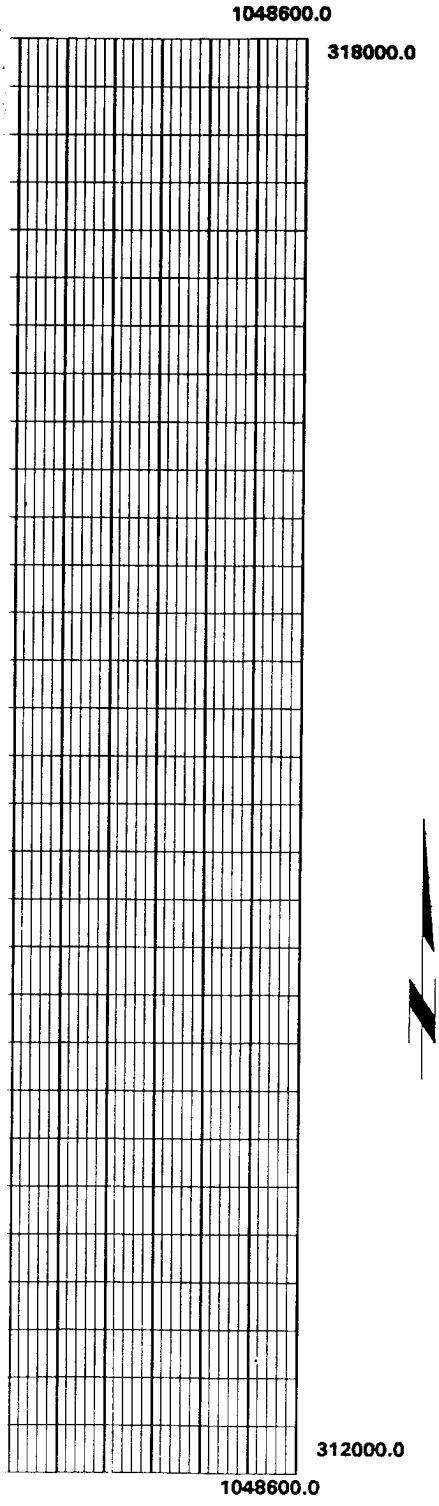
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- **Single Targets**
- **Light Density Target Areas**
- **Trench Areas**
- **Surface Anomalies**
- **Scattered Debris**
- **High Density Target Areas**


1042600.0
318000.0
312000.0
1042600.0



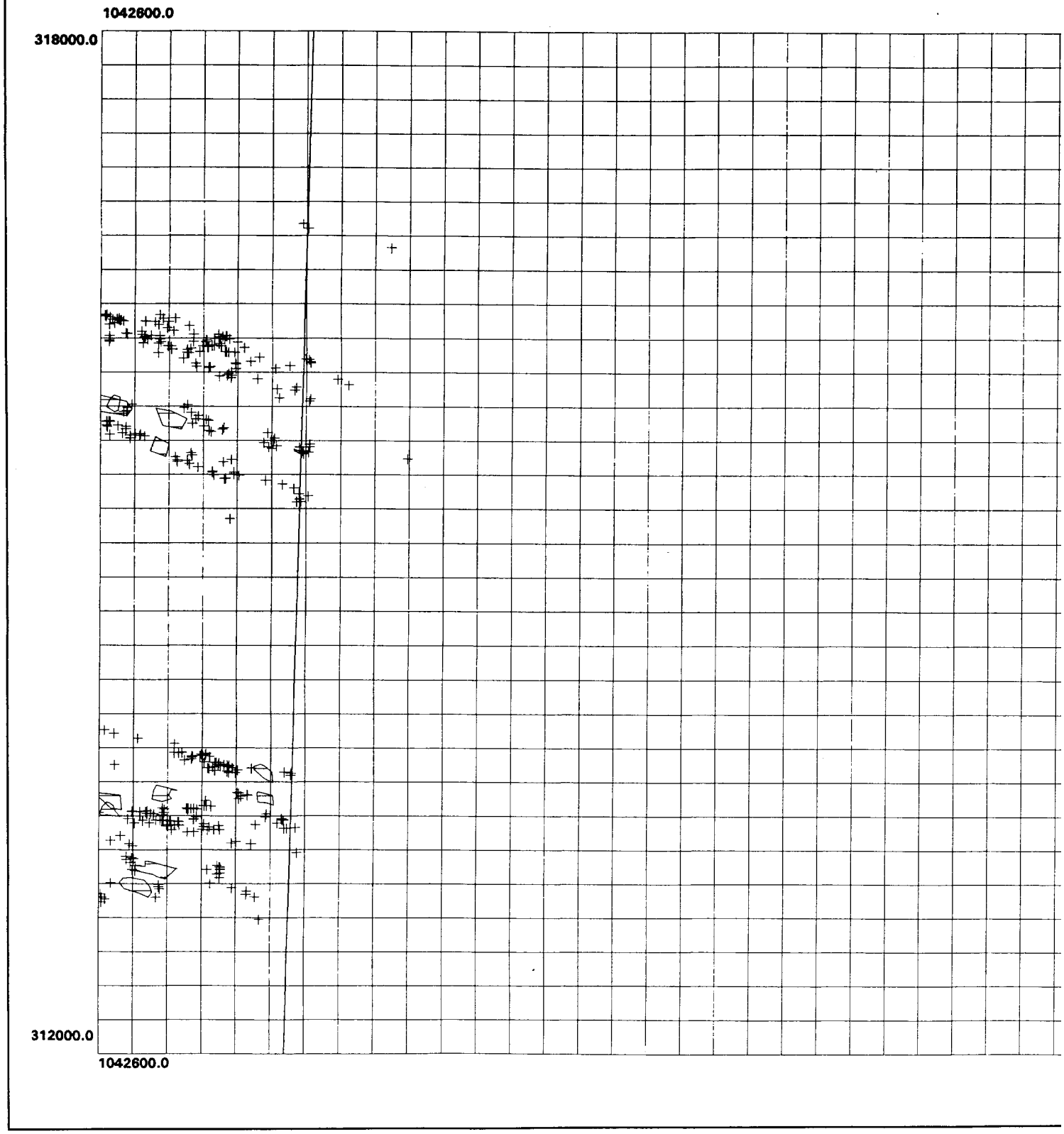
M13 West Burning Ground 2
4 Satellites Available
Test Flight 1, Test Flight 2,
Test Flight 4, &
Test Flight 7

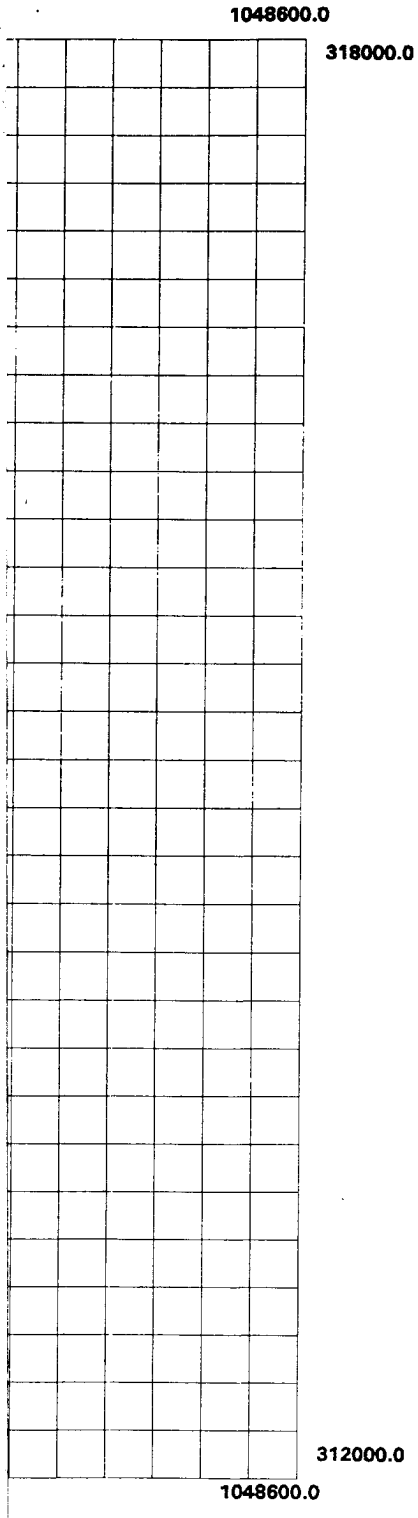
Release Date:
14 Oct 1994

800 0 800 1600 Feet

Legend

- Single Targets
- Light Density Target Areas
- Trench Areas
- Surface Anomalies
- Scattered Debris
- High Density Target Areas
- Holiday Areas
- Digital Imagery


1042600.0
318000.0
312000.0
1042600.0



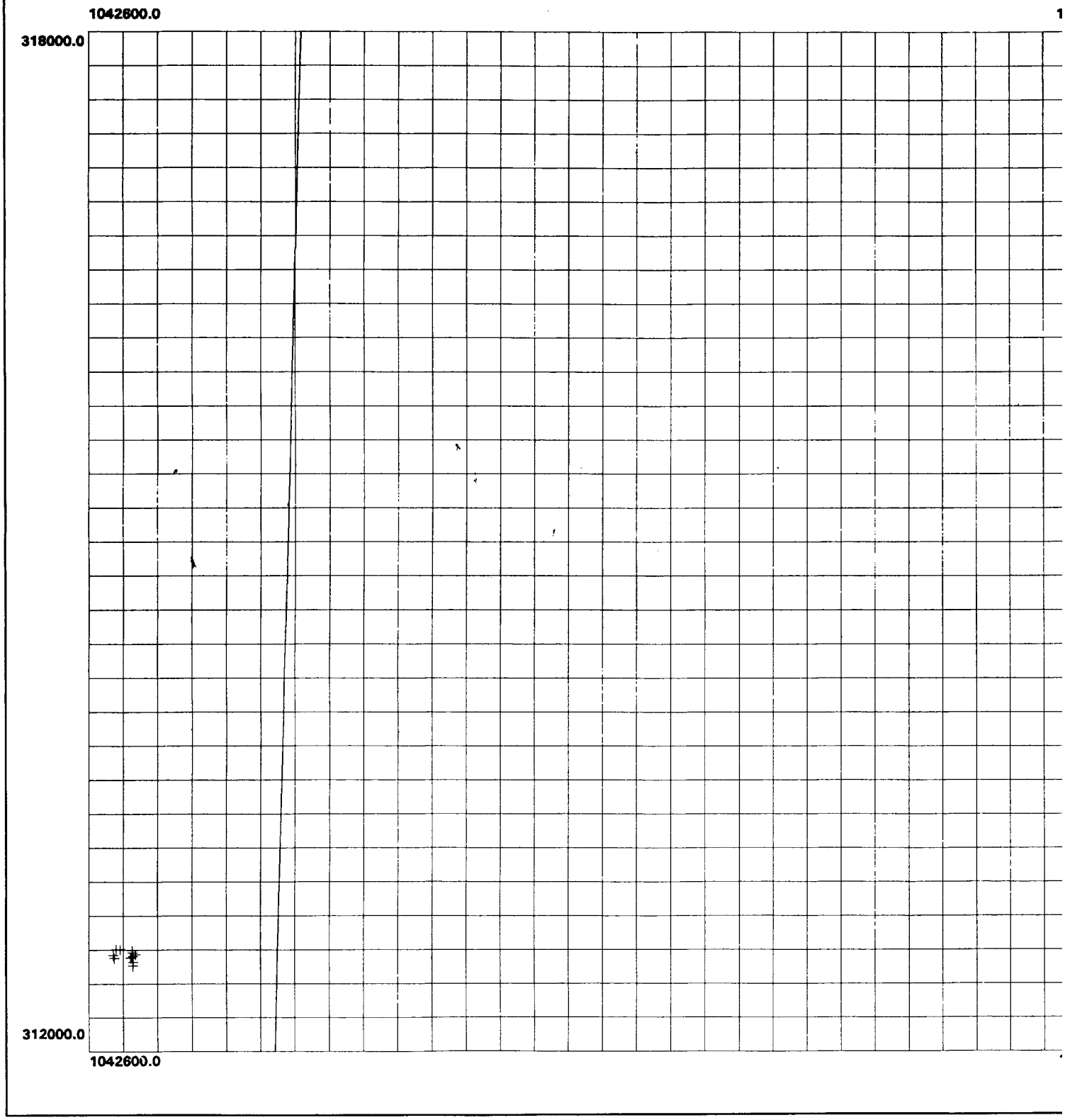
West Burning Ground 2
3 Satellites Available
Test Flight 1, Test Flight 2,
Test Flight 4, &
Test Flight 7

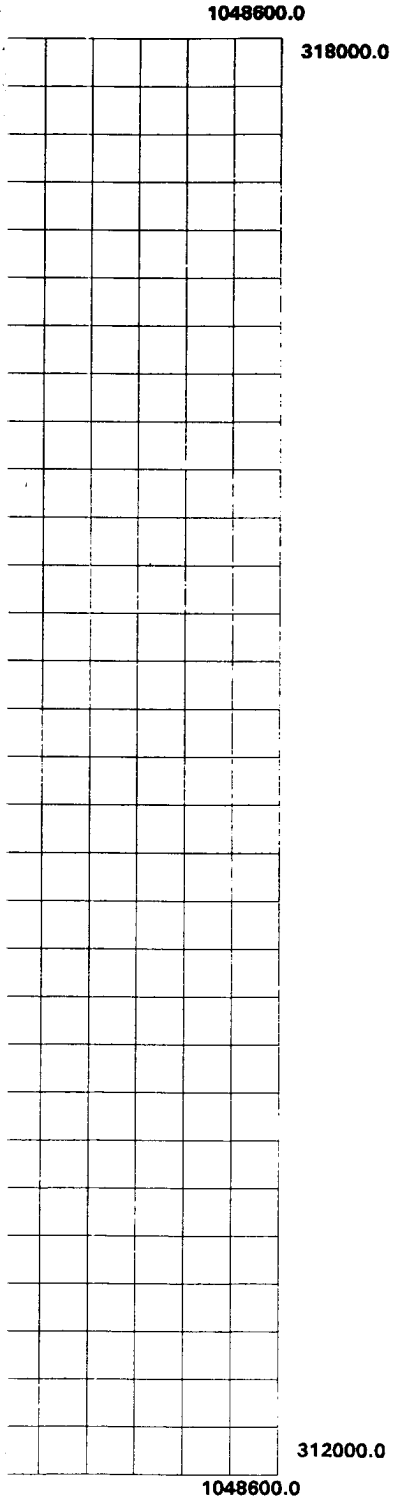
Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- **Single Targets**
- **Light Density Target Areas**
- **Trench Areas**
- **Surface Anomalies**
- **Scattered Debris**
- **High Density Target Areas**


1042600.0
318000.0
312000.0
1042600.0



West Burning Ground 2
Uncorrectable GPS Coverage
Test Flight 2 ONLY

Release Date:
03 Dec 1993

800 0 800 1600 Feet

Legend

- **Single Targets**
- **Light Density Target Areas**
- **Trench Areas**
- **Surface Anomalies**
- **Scattered Debris**
- **High Density Target Areas**

Appendix B
ARC/INFO GIS Macro Listings

ARC/INFO Macro code listings for the Black Hills Army Depot map generation are given in this appendix.

The following listings were used to generate the four-satellite coverage maps.

F1_74.AML Listing

```
/* F1_74.AML
/*
/* Generates All Flights aml
/*    for 4 Satellite Coverage
/*    of Full Size Maps
/*
&setvar pprog = fltpp4
&setvar mscale = 2400
&run f454b1.aml %pprog% %mscale%
cp plotbhad.gra f454b1.gra
&run f454eb1.aml %pprog% %mscale%
cp plotbhad.gra f454eb1.gra
&run f124674b2bd.aml %pprog% %mscale%
cp plotbhad.gra f124674b2bd.gra
&run f12474wb2.aml %pprog% %mscale%
cp plotbhad.gra f12474wb2.gra
&run f344b3.aml %pprog% %mscale%
cp plotbhad.gra f344b3.gra
&run f344sb3.aml %pprog% %mscale%
cp plotbhad.gra f344sb3.gra
&run f34c.aml %pprog% %mscale%
cp plotbhad.gra f34c.gra
&run f34t.aml %pprog% %mscale%
cp plotbhad.gra f34t.gra
&run f64bd.aml %pprog% %mscale%
cp plotbhad.gra f64bd.gra
&run f64bf.aml %pprog% %mscale%
cp plotbhad.gra f64bf.gra
&run f74bg.aml %pprog% %mscale%
cp plotbhad.gra f74bg.gra
&run f64dp.aml %pprog% %mscale%
cp plotbhad.gra f64dp.gra
&run f64b8.aml %pprog% %mscale%
cp plotbhad.gra f64b8.gra
```

F1_7S4.AML Listing

```
/* F1_7S4.AML
/*
/* Generates All Flights aml
/*    for 4 Satellite Coverage
/*    of Working Size Maps
/*
&setvar pprog = fltpps4
&setvar mscale = 9600
&run f454b1.aml %pprog% %mscale%
cp plotbhad.gra f454b1s.gra
&run f454eb1.aml %pprog% %mscale%
cp plotbhad.gra f454eb1s.gra
&run f124674b2bd.aml %pprog% %mscale%
cp plotbhad.gra f124674b2bds.gra
&run f12474wb2.aml %pprog% %mscale%
cp plotbhad.gra f12474wb2s.gra
&run f344b3.aml %pprog% %mscale%
cp plotbhad.gra f344b3s.gra
&run f344sb3.aml %pprog% %mscale%
cp plotbhad.gra f344sb3s.gra
&run f34c.aml %pprog% %mscale%
cp plotbhad.gra f34cs.gra
&run f34t.aml %pprog% %mscale%
cp plotbhad.gra f34ts.gra
```

```
&run f64bd.aml %pprog% %mscale%
cp plotbhad.gra f64bds.gra
&run f64bf.aml %pprog% %mscale%
cp plotbhad.gra f64bfs.gra
&run f74bg.aml %pprog% %mscale%
cp plotbhad.gra f74bgs.gra
&run f64dp.aml %pprog% %mscale%
cp plotbhad.gra f64dps.gra
&run f64b8.aml %pprog% %mscale%
cp plotbhad.gra f64b8s.gra
```

F454B1.AML Listing

```
&args pprog mscale
cp f454b1.txt fltpp.txt
&run %pprog% %mscale% 1042600.0 318000.0 1049400.0 324000.0 none flt514+flt524+flt414+flt424+flt434+a07f45b
```

F454B1.TXT Listing

```
M10  Burning Ground 1
4 Satellites Available
Test Flight 4 &
Test Flight 5
```

F454EB1.AML Listing

```
&args pprog mscale
cp f454eb1.txt fltpp.txt
&run %pprog% %mscale% 1049400.0 318000.0 1056200.0 324000.0 none flt514+flt524+flt414+flt424+flt434+a07f45b
```

F454EB1.TXT Listing

```
M11  East of Burning Ground 1
4 Satellites Available
Test Flight 4 &
Test Flight 5
```

F124674B2BD.AML Listing

```
&args pprog mscale
cp f124674b2bd.txt fltpp.txt
&run %pprog% %mscale% 1036600.0 312000.0 1042600.0 318000.0 none
flt124+flt21a4+flt21g4+flt22g4+flt23g4+flt24g4+flt414+flt614+flt744+a07f12467b
```

F124674B2BD.TXT Listing

```
M12  Burning Ground 2 & Block D
4 Satellites Available
Test Flight 1, Test Flight 2,
Test Flight 4, Test Flight 6, &
Test Flight 7
```

F12474WB2.AML Listing

```
&args pprog mscale
cp f12474wb2.txt fltpp.txt
&run %pprog% %mscale% 1042600.0 312000.0 1048600.0 318000.0 none
flt124+flt21a4+flt21g4+flt22g4+flt23g4+flt24g4+flt414+flt744+a07f12467b
```

F12474WB2.TXT Listing

```
M13  West Burning Ground 2
4 Satellites Available
```

Test Flight 1, Test Flight 2,
Test Flight 4, &
Test Flight 7

F344B3.AML Listing

```
&args pprog mscale
cp f344b3.txt fltpp.txt
&run %pprog% %mscale% 1048000.0 333600.0 1054000.0 339600.0 none
flt314+flt324+flt334+flt414+flt424+flt434+a07f34b
```

F344B3.TXT Listing

```
M04  Burning Ground 3
4 Satellites Available
Test Flight 3 &
Test Flight 4
```

F344SB3.AML Listing

```
&args pprog mscale
cp f344sb3.txt fltpp.txt
&run %pprog% %mscale% 1048000.0 327600.0 1054000.0 333600.0 none
flt314+flt324+flt334+flt414+flt424+flt434+a07f34b
```

F344SB3.TXT Listing

```
M05  South Burning Ground 3
4 Satellites Available
Test Flight 3 &
Test Flight 4
```

F34C.AML Listing

```
&args pprog mscale
cp f34c.txt fltpp.txt
&run %pprog% %mscale% 1029600.0 336000.0 1035600.0 342000.0 none flt314+flt324+flt334+a07f3b
```

F34C.TXT Listing

```
M01  Chemical Site
4 Satellites Available
Test Flight 3
```

F34T.AML Listing

```
&args pprog mscale
cp f34t.txt fltpp.txt
&run %pprog% %mscale% 1036000.0 337000.0 1042000.0 343000.0 none flt314+flt324+flt334+a07f3b
```

F34T.TXT Listing

```
M02  Target Range
4 Satellites Available
Test Flight 3
```

F64BD.AML Listing

```
&args pprog mscale
cp f64bd.txt fltpp.txt
&run %pprog% %mscale% 1036600.0 318000.0 1042600.0 324000.0 none flt614+a07f6b
```

F64BD.TXT Listing

```
M09  Block D
4 Satellites Available
Test Flight 6
```

F64BF.AML Listing

```
&args pprog mscale
cp f64bf.txt fltpp.txt
&run %pprog% %mscale% 1038800.0 324000.0 1044800.0 329600.0 none flt6224+flt634+flt644+a07f6b
```

F64BF.TXT Listing

```
M08  Block F
4 Satellites Available
Test Flight 6
```

F74BG.AML Listing

```
&args pprog mscale
cp f74bg.txt fltpp.txt
&run %pprog% %mscale% 1034800.0 329600.0 1040800.0 335600.0 none flt714+flt724+a07f7b
```

F74BG.TXT Listing

```
M03  Block G
4 Satellites Available
Test Flight 7
```

F64DP.AML Listing

```
&args pprog mscale
cp f64dp.txt fltpp.txt
&run %pprog% %mscale% 1052600.0 326200.0 1058600.0 332200.0 none flt6244+a07f6b
```

F64DP.TXT Listing

```
M06  Disassembly Plant Area
4 Satellites Available
Test Flight 6
```

F64B8.AML Listing

```
&args pprog mscale
cp f64b8.txt fltpp.txt
&run %pprog% %mscale% 1058600.0 326200.0 1064600.0 332200.0 none flt614+flt6224+flt6244+a07f6b
```

F64B8.TXT Listing

```
M07  Building 8000 Area
4 Satellites Available
Test Flight 6
```

FLTPP4.AML Listing

```
/* FLTPP4.AML
/*
/* Flight Plot Processing aml
/*    for 4 Satellite Coverage
/*    of Full Size Maps
/*
```

```
&args mscale minx miny maxx maxy gandfs:REST
&if [null %gandfs%] &then &goto usage

/* start arcplot if args are OK
arcplot

&setvar munlen = %mscale% / 12
&setvar mapbor = 0.5
&setvar mmaxx = [calc ( ( %maxx% - %minx% ) / %munlen% ) + %mapbor%]
&setvar mmaxy = [calc ( ( %maxy% - %miny% ) / %munlen% ) + %mapbor%]
&setvar linesyms = '41+49'
&setvar gpss = [before %gandfs% ' ']
&setvar flts = [after %gandfs% ' ']
&type %mscale% %munlen% %mapbor% %mmaxx% %mmaxy%
&type %gpss% %linesyms% %flts%

display 1040
plotbhad
mape %minx% %miny% %maxx% %maxy%
mapunits feet
mapscale %mscale%
maplimits 0.5 0.5 %mmaxx% %mmaxy%

/* Plot the Base Map Arcs
resel cc314 arcs cc314-id = 100
arcs cc314

/* Plot the 200 ft. by 200 ft. grid (fish net)
arcs bhadfn

/* Plot GPS Data if given
markerset plotter
lineset plotter
shadeset plotter
linesize 0.04
&if %gpss% = 'none' &then &goto flights
&do &while not [nul %gpss%]
  &setvar gps = [before %gpss% +]
  &setvar gpss = [after %gpss% +]
  &setvar linesym = [before %linesyms% +]
  &setvar linesyms = [after %linesyms% +]
  clearsel
  linesymbol %linesym%
  resel %gps% arcs %gps%-id = 1
  arcs %gps%
&end

/* Plot Flight Data if given
&label flights
&if %flts% = 'none' &then &goto legend
&do &while not [nul %flts%]
  &setvar flt = [before %flts% +]
  &setvar flts = [after %flts% +]
  &if [exists %flt%pts -cover] &then
  &do
    clearsel
    markersymbol 1
    resel %flt%pts points target = 1
    points %flt%pts
  &end

/* Use Dissolved Coverage if present
  &setvar fltnd = 'none'
  &setvar fltd = %flt%d
  &if [exists %fltd%ply -cover] &then
  &do
```

```
    &setvar fltnd = %flt%
    &setvar flt = %fltd%
  &end

/* Plot Target Polys
  &if [exists %flt%ply -cover] &then
  &do
    clearsel
    resel %flt%ply polys target = 2
    polygonshades %flt%ply 5
    clearsel
    resel %flt%ply polys target = 3
    polygonshades %flt%ply 65
    clearsel
    resel %flt%ply polys target = 4
    polygonshades %flt%ply 9
    clearsel
    resel %flt%ply polys target = 5
    polygonshades %flt%ply 81
    clearsel
    resel %flt%ply polys target = 6
    polygonshades %flt%ply 61
    linesymbol 1
    clearsel
    arcs %flt%ply
  &end

/* Plot Holiday Areas
  &if [exists %flt% -cover] &then
  &do
    clearsel
    resel %flt% polys inside = 1
    polygonshades %flt% 21
    linesymbol 5
    clearsel
    arcs %flt%
  &end

/* Plot Arcs From Non-Dissolved Coverage
  &if [exists %fltnd%ply -cover] &then
  &do
    clearsel
    linesymbol 1
    clearsel
    arcs %fltnd%ply
  &end
&end

/* Plot Digital Imagery Locations
    clearsel
    resel m4allply polys target = 8
    polygonshades m4allply 17
    linesymbol 1
    clearsel
    arcs m4allply

/* Label Digital Imagery
textquality proportional
textfont 94023
textsize .2 .175
labeltext m4allply label # CC NOROTATION

/* Print Map Title
&label legend
textquality proportional
textfont 94023
```

```
textsize .2 .175
move [calc %mmaxx% + 1.0] [calc %mmaxy% - 2.0]
textfile fltpp.txt

/* Print Release Date
move [calc %mmaxx% + 4.5] [calc %mmaxy% - 2.0]
textfile reldate4.txt

/* Print Legend
&setvar legoffset = 9.2
move [calc %mmaxx% + 3.0] [calc %mmaxy% - %legoffset%]
text 'Legend'
&setvar legoffset = [calc %legoffset% + 0.3]
linesymbol 1
keyposition [calc %mmaxx% + 2.0] [calc %mmaxy% - %legoffset%]
keyseparation .15 .15
keymarker fltpppts.leg nobox
&setvar legoffset = [calc %legoffset% + 0.5]
&if %flts% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 2.0] [calc %mmaxy% - %legoffset%]
  keyshade fltppply.leg
  keyposition [calc %mmaxx% + 4.0] [calc %mmaxy% - %legoffset%]
  keyshade fltply4.leg
  &setvar legoffset = [calc %legoffset% + 3.5]
&end
&if %gpss% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 2.0] [calc %mmaxy% - %legoffset%]
  keyline fltppline.leg
&end

/* Print Scale Bar, North Arrow, WES Logo, and Castle
linesymbol 5
&run scalebar [calc %mmaxx% + 3.0] [calc %mmaxy% - 7.5] FEET %munlen% 2 3
northarrow [calc %mmaxx% + 0.5] [calc %mmaxy% - 11.5] 1.5
&run bweslogo [calc %mmaxx% + 3.5] [calc %mmaxy% - 4.5] 1.5
&run bcastle [calc %mmaxx% + 1.0] [calc %mmaxy% - 4.25] 1.5

/* draw borders
box 0 0 [calc %mmaxx% + 6.5] [calc %mmaxy% + 0.5]
box %mapbor% %mapbor% %mmaxx% %mmaxy%

/* write out map corners
/* lower left
move 0.5 0.4
text %minx%
move 0.01 0.6
text %miny%
/* lower right
move [calc %mmaxx% - 0.4] 0.4
text %maxx%
move [calc %mmaxx% + 0.1] 0.6
text %miny%
/* upper left
move 0.5 [calc %mmaxy% + 0.1]
text %minx%
move 0.01 [calc %mmaxy% - 0.1]
text %maxy%
/* upper right
move [calc %mmaxx% - 0.4] [calc %mmaxy% + 0.1]
text %maxx%
move [calc %mmaxx% + 0.1] [calc %mmaxy% - 0.1]
text %maxy%

/* quit arcplot
```

```
quit
&return
&end

/*
/*  fltpp4 usage
/*
&label usage
  &type Usage: FLTPP4 <mapscale> <minx> <maxx> <maxy>
  &type              <gpscover1>...<{+gpscovern} <fltcover1>...{+fltcovern}
  &return
```

FLTPPS4.AML Listing

```
/*  FLTPPS4.AML
/*
/*  Flight Plot Processing Small format aml
/*     for 4 Satellite Coverage
/*     of Working Size Maps
/*
&args mscale minx miny maxx maxy gandfs:REST
&if [null %gandfs%] &then &goto usage

/* start arcplot if args are OK
arcplot

&setvar munlen = %mscale% / 12
&setvar mapbor = 0.5
&setvar mmaxx = [calc ( ( %maxx% - %minx% ) / %munlen% ) + %mapbor%]
&setvar mmaxy = [calc ( ( %maxy% - %miny% ) / %munlen% ) + %mapbor%]
&setvar linesyms = '41+49'
&setvar gpss = [before %gandfs% ' ']
&setvar flts = [after %gandfs% ' ']
&type %mscale% %munlen% %mapbor% %mmaxx% %mmaxy%
&type %gpss% %linesyms% %flts%

display 1040
plotbhad
mape %minx% %miny% %maxx% %maxy%
mapunits feet
mapscale %mscale%
maplimits 0.5 0.5 %mmaxx% %mmaxy%

/*  Plot the Base Map Arcs
resel cc314 arcs cc314-id = 100
arcs cc314

/*  Plot the 200 ft. by 200 ft. grid (fish net)
arcs bhadfn

/*  Plot GPS Data if given
markerset plotter
lineset plotter
shadeset plotter
linesize 0.04
&if %gpss% = 'none' &then &goto flights
&do &while not [nul %gpss%]
  &setvar gps = [before %gpss% +]
  &setvar gpss = [after %gpss% +]
  &setvar linesym = [before %linesyms% +]
  &setvar linesyms = [after %linesyms% +]
  clearsel
  linesymbol %linesym%
  resel %gps% arcs %gps%-id = 1
```

```
  arcs %gps%
&end

/* Plot Flight Data if given
&label flights
&if %flts% = 'none' &then &goto legend
&do &while not [nul %flts%]
  &setvar flt = [before %flts% +]
  &setvar flts = [after %flts% +]
  &if [exists %flt%pts -cover] &then
  &do
    clearsel
    markersymbol 1
    resel %flt%pts points target = 1
    points %flt%pts
  &end

/* Use Dissolved Coverage if present
  &setvar fltnd = 'none'
  &setvar fltd = %flt%d
  &if [exists %fltd%ply -cover] &then
  &do
    &setvar fltnd = %flt%
    &setvar flt = %fltd%
  &end

/* Plot Target Polys
  &if [exists %flt%ply -cover] &then
  &do
    clearsel
    resel %flt%ply polys target = 2
    polygonshades %flt%ply 5
    clearsel
    resel %flt%ply polys target = 3
    polygonshades %flt%ply 65
    clearsel
    resel %flt%ply polys target = 4
    polygonshades %flt%ply 9
    clearsel
    resel %flt%ply polys target = 5
    polygonshades %flt%ply 81
    clearsel
    resel %flt%ply polys target = 6
    polygonshades %flt%ply 61
    linesymbol 1
    clearsel
    arcs %flt%ply
  &end

/* Plot Holiday Areas
  &if [exists %flt% -cover] &then
  &do
    clearsel
    resel %flt% polys inside = 1
    polygonshades %flt% 21
    linesymbol 5
    clearsel
    arcs %flt%
  &end

/* Plot Arcs From Non-Dissolved Coverage
  &if [exists %fltnd%ply -cover] &then
  &do
    clearsel
    linesymbol 1
    clearsel
```

```
    arcs %fltnd%ply
  &end
&end

/* Draw Digital Imagery Locations
    clearsel
    resel m4allply polys target = 8
    polygonshades m4allply 17
    linesymbol 1
    clearsel
    arcs m4allply

/* Label Digital Imagery
textquality proportional
textfont 94023
textsize .075 .06
labeltext m4allply label # CC NOROTATION

/* Print Map Title
&label legend
textquality proportional
textfont 94023
textsize .2 .175
move [calc %mmaxx% + 1.0] [calc %mmaxy% - 0.0]
textfile fltpp.txt

/* Print Release Date
move [calc %mmaxx% + 4.5] [calc %mmaxy% - 0.0]
textfile reldate4.txt

/* Print Legend
&setvar legoffset = 3.2
move [calc %mmaxx% + 3.0] [calc %mmaxy% - %legoffset%]
text 'Legend'
&setvar legoffset = [calc %legoffset% + 0.3]
linesymbol 1
keyposition [calc %mmaxx% + 2.0] [calc %mmaxy% - %legoffset%]
keyseparation .15 .15
keymarker fltpppts.leg nobox
&setvar legoffset = [calc %legoffset% + 0.5]
&if %flts% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 2.0] [calc %mmaxy% - %legoffset%]
  keyshade fltppply.leg
  keyposition [calc %mmaxx% + 4.0] [calc %mmaxy% - %legoffset%]
  keyshade fltply4.leg
  &setvar legoffset = [calc %legoffset% + 3.5]
&end
&if %gpss% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 2.0] [calc %mmaxy% - %legoffset%]
  keyline fltppline.leg
&end

/* Print Scale Bar, North Arrow, WES Logo, and Castle
linesymbol 5
&run scalebar [calc %mmaxx% + 3.0] [calc %mmaxy% - 2.5] FEET %munlen% 2 3
northarrow [calc %mmaxx% + 0.5] [calc %mmaxy% - 5.5] 1.5
&run bweslogo [calc %mmaxx% + 3.5] [calc %mmaxy% - 2.0] 1.5
&run bcastle [calc %mmaxx% + 1.0] [calc %mmaxy% - 1.75] 1.5

/* draw borders
box 0 0 [calc %mmaxx% + 6.5] [calc %mmaxy% + 0.5]
box %mapbor% %mapbor% %mmaxx% %mmaxy%

/* write out map corners
```

```
/* lower left
move 0.5 0.4
text %minx%
move 0.01 0.6
text %miny%
/* lower right
move [calc %mmaxx% - 0.4] 0.4
text %maxx%
move [calc %mmaxx% + 0.1] 0.6
text %miny%
/* upper left
move 0.5 [calc %mmaxy% + 0.1]
text %minx%
move 0.01 [calc %mmaxy% - 0.1]
text %maxy%
/* upper right
move [calc %mmaxx% - 0.4] [calc %mmaxy% + 0.1]
text %maxx%
move [calc %mmaxx% + 0.1] [calc %mmaxy% - 0.1]
text %maxy%

/* quit arcplot
quit
&return
&end

/*
/*  fltpps4 usage
/*
&label usage
  &type Usage: FLTPP4 <mapscale> <minx> <maxx> <maxy>
  &type              <gpscover1>...<{+gpscovern} <fltcover1>...{+fltcovern}
  &return
```

RELDATE4.TXT Listing

```
Release Date:
14 Oct 1994
```

FLTPPPTS.LEG Listing

```
.1
Single
Targets
```

FLTPPPLY.LEG Listing

```
.5
Light Density
Target Areas
.65
Trench
Areas
.9
Surface
Anomalies
.81
Scattered
Debris
.61
High Density
Target Areas
```

FLTPLY4.LEG Listing

```
.21
Holiday
Areas
.17
Digital
Imagery
```

FLTPPLINE.LEG Listing

```
.41
Test Flight w/ 4 Satellites
.49
Test Flight w/ 3 Satellites
```

SCALEBAR.AML Listing

```
/* SCALEBAR.AML
/*
/* Description: This AML Draws a scale bar for a map displayed in ARC/INFO.
/*
/*
/* Author:     R. Eddie Melton, Jr
/*           Atlantic Research Corporation
/*           US Army Corps of Engineers
/*           Waterways Experiment Station
/*           Vicksburg, MS
/*
/*
/* Arguments:
/*
/*    xpos     (required) - x coordinate
/*           This is the x coordinate location for the center of the
/*           scale or the position of the 0 specified by the location
/*           argument.
/*
/*    ypos     (required) - y coordinate
/*           This is the y coordinate location for the center of the
/*           scale.
/*
/*    s_units  (required) - Scale units
/*           This is the units to be represented by the scale.  The
/*           valid units are:
/*             KM
/*             MILE
/*             METER
/*             FEET
/*
/*    inc_val  (required) - Increment value
/*           This is the number of units to be represented by one
/*           segment of the scalebar.
/*
/*    n_inc    (required) - Number of increments
/*           This is the number of segments to be placed in the scalebar.
/*
/*    n_div    (required) - Number of divisions
/*           This is the number of divisions to be placed in the segment
/*           to the left of 0.
/*
/*    location (optional) - Location for xpos and ypos
/*           This is where the xpos and ypos coordinates are to be
/*           placed with respect to the scalebar.  The valid locations
/*           are:
/*             CENTER (Default)
```

```
/*              POS0
/*
/*    filler   (optional)  - Kind of filler
/*             This is the type of filler to be used to fill every other
/*             segment.  The valid filler values are:
/*              SOLID (Default)
/*              LINE
/*

&ARGS xpos ypos s_units inc_val n_inc n_div location filler

&if [null %xpos%] &then &goto usage
&if [null %ypos%] &then &goto usage
&if [null %s_units%] &then &goto usage
&if [null %inc_val%] &then &goto usage
&if [null %n_inc%] &then &goto usage
&if [null %n_div%] &then &goto usage
&if [null %location%] &then &setvar location CENTER
&if [null %filler%] &then &setvar filler SOLID


textquality proportional
textfont 94023
lineset plotter.lin
shadeset plotter.shd

&setvar scale := [show mapscale]
/*
/* uncomment this when the code is modified to work with different page units
/*&setvar units := [show pageunits]
/*
&setvar units := INCHES

/*
/* this is the conversion values used with inches
/*
&setvar KM := 39370.07874
&setvar MILE := 63360
&setvar FEET := 12
&setvar METER := 39.37007874
&setvar INCH := 1

/*
/* find the length of one segment
/*
&setvar UNITLENGTH := %inc_val% * [value %s_units%] / %scale%

/*
/* Check for the location to place xpos and ypos
/*
&if %location% = 'POS0' &then
  &setvar LLX := %xpos% - %UNITLENGTH%
&if %location% = 'CENTER' &then
  &setvar LLX := %xpos% - ( %UNITLENGTH% * ( %n_inc% + 1 ) / 2 )

/*
/* Calculate the remaining boundaries of the scale
/*
&setvar URX := %LLX% + ( %UNITLENGTH% * ( %n_inc% + 1 ) )
&setvar LLY := %ypos% - 0.05
&setvar URY := %ypos%

/*
/* Draw a box for the scalebar
/*
box %LLX% %LLY% %URX% %URY%
```

```
/*
/* Draw the vertical lines for each segment
/*
&setvar x1 := %LLX%
&do count := 1 &to %n_inc% &by 1
  &setvar x1 := %x1% + %UNITLENGTH%
  line %x1% %LLY% %x1% %URY%
&end

/*
/* Draw the vertical lines for each division
/*
&setvar x1 := %LLX%
&do count := 1 &to %n_div% &by 1
  &setvar x1 := %x1% + ( %UNITLENGTH% / %n_div% )
  line %x1% %LLY% %x1% %URY%
&end

/*
/* If the filler is LINE, draw lines for the segments and divisions
/*
&if %filler% = 'LINE' &then &do
  &setvar x1 := %LLX%
  &setvar y1 := %LLY% + 0.025
  &do count := 1 &to %n_div% &by 2
    &setvar x2 := %x1% + ( %UNITLENGTH% / %n_div% )
    line %x1% %y1% %x2% %y1%
    &setvar x1 := %x2% + ( %UNITLENGTH% / %n_div% )
  &end
  &setvar x1 := %LLX% + ( %UNITLENGTH% * 2 )
  &do count := 2 &to %n_inc% &by 2
    &setvar x2 := %x1% + %UNITLENGTH%
    line %x1% %y1% %x2% %y1%
    &setvar x1 := %x2% + %UNITLENGTH%
  &end
&end

/*
/* If the filler is SOLID, patch blocks for the segments and divisions
/*
&if %filler% = 'SOLID' &then &do
  shadesymbol 1
  &setvar x1 := %LLX%
  &setvar y1 := %LLY%
  &setvar y2 := %URY%
  &do count := 1 &to %n_div% &by 2
    &setvar x2 := %x1% + ( %UNITLENGTH% / %n_div% )
    patch %x1% %y1% %x2% %y2%
    &setvar x1 := %x2% + ( %UNITLENGTH% / %n_div% )
  &end
  &setvar x1 := %LLX% + ( %UNITLENGTH% * 2 )
  &do count := 2 &to %n_inc% &by 2
    &setvar x2 := %x1% + %UNITLENGTH%
    patch %x1% %y1% %x2% %y2%
    &setvar x1 := %x2% + %UNITLENGTH%
  &end
&end

/*
/* Set the text size for the measurements
/*
textsize 0.100

/*
/* Place a 0 at the second vertical line
```

```
/*
&setvar y := %URY% + 0.05
&setvar x := %LLX% + %UNITLENGTH%
&setvar scaletxt := 0
move %x% %y%
text %scaletxt% cc

/*
/* Place the first incremental value to the left of 0
/*
&setvar x := %x% - %UNITLENGTH%
&setvar scaletxt := %scaletxt% + %inc_val%
move %x% %y%
text %scaletxt% cc

/*
/* Place the first incremental value to the right of 0
/*
&setvar x := %x% + %UNITLENGTH% + %UNITLENGTH%
move %x% %y%
text %scaletxt% cc

/*
/* Place the remaining increments at each vertical line
/*
&do count := 2 &to %n_inc% &by 1
  &setvar x := %x% + %UNITLENGTH%
  &setvar scaletxt := %scaletxt% + %inc_val%
  move %x% %y%
  text %scaletxt% cc
&end

/*
/* Place the type of measurement at the right of the scale
/*
&setvar x := %x% + 0.15
move %x% %y%
&if %s_units% = 'KM' &then
  text 'Kilometers' cl
&if %s_units% = 'MILE' &then
  text 'Statute Miles' cl
&if %s_units% = 'FEET' &then
  text 'Feet' cl
&if %s_units% = 'METER' &then
  text 'Meters' cl


&return
&end


/*
/* Display usage
/*
&label usage
  &type Usage: SCALEBAR <xpos> <Ypos> <scale units> <increment>
  &type                <number segments> <number divisions>
  &type                {coordinate location}  {fill type}

  &return
```

BWESLOGO.AML Listing

```
/*
/* Plot WES logo at xorigin,yorigin
/* The logo is %size% mapunits
/*
&ARGS xorigin yorigin size

&if [null %xorigin%] &then &goto usage
&if [null %yorigin%] &then &goto usage
&if [null %size%] &then &goto usage

mapscale automatic

&sv location := /local/arcinfo/logos/

&sv lwidth := %xorigin% + %size%
&sv lheight := %yorigin% + %size%

mapex %location%weslogo

shadeset plotter.shd
/* linecolor white
lineset plotter.lin
linesymbol 1


maplimits %xorigin% %yorigin% %lwidth% %lheight%

/* clearselect
/* reselect %location%weslogo polys weslogo-id = 26
/* polygonshades %location%weslogo 37

clearselect
reselect %location%weslogo polys weslogo-id = 2
polygonshades %location%weslogo 85

clearselect
reselect %location%weslogo polys weslogo-id = 3
polygonshades %location%weslogo 73

clearselect
reselect %location%weslogo polys weslogo-id = 4
polygonshades %location%weslogo 1

clearselect
reselect %location%weslogo arcs weslogo-id ne 888888
arcs %location%weslogo
clearselect

&return
&end


/*
/* Display usage
/*
&label usage
  &type Usage: BWESLOGO <xorigin> <yorigin> <size(pageunits)>
  &return
```

BCASTLE.AML Listing

```
/*
/*  Plot USACE logo at xorigin,yorigin
/*  The logo is %size% mapunits
/*
&ARGS xorigin yorigin size

&if [null %xorigin%] &then &goto usage

mapscale automatic

&sv location := /local/arcinfo/logos/

&sv lwidth := %xorigin% + %size%
&sv lheight := %yorigin% + %size%

mapex arcs %location%castle

shadeset plotter.shd
lineset plotter.lin
linesymbol 1

maplimits %xorigin% %yorigin% %lwidth% %lheight%

/* clearselect
/* reselect %location%castle polys castle-id = 1
/* polygonshades %location%castle 73

clearselect
reselect %location%castle polys castle-id = 2
polygonshades %location%castle 85

clearselect
linesymbol 1
arcs %location%castle

&return
&end


/*
/* Display usage
/*
&label usage
  &type Usage: BCASTLE<xorigin> <yorigin> <size(pageunits)>
  &return
```

P1_74.AML Listing

```
rasplot f454b1.gra
rasplot f454eb1.gra
rasplot f124674b2bd.gra
rasplot f12474wb2.gra
rasplot f344b3.gra
rasplot f344sb3.gra
rasplot f34c.gra
rasplot f34t.gra
rasplot f64bd.gra
rasplot f64bf.gra
rasplot f74bg.gra
rasplot f64dp.gra
rasplot f64b8.gra
```

P1_7S4.AML Listing

```
rasplot f454b1s.gra
rasplot f454eb1s.gra
rasplot f124674b2bds.gra
rasplot f12474wb2s.gra
rasplot f344b3s.gra
rasplot f344sb3s.gra
rasplot f34cs.gra
rasplot f34ts.gra
rasplot f64bds.gra
rasplot f64bfs.gra
rasplot f74bgs.gra
rasplot f64dps.gra
rasplot f64b8s.gra
```

RASPLOT.AML Listing

```
/*********************************************************
/*
/* NAME: rasplot.aml
/*
/* PURPOSE: Spools an HPGL-2 or Calcomp 68000 file
/*        to the Rastergraphics Electrostatic Plotter.
/*
/* AUTHOR: Mark R. Graves
/*       US Army Engineer Waterways Experiment Station
/*       Environmental Laboratory
/*       3909 Halls Ferry Road (WESEN-C)
/*       Vicksburg, MS 39180
/*       (601)634-2557
/*
/* DATE:  1/92 Original Coding  (for Versatec plotter)
/*      2/92 Added Usage response
/*      6/92 Added option to remove banner  (Default = BANNER)
/*      8/92 Created versatec2 to support second versatec
/*      3/93 Added color calibration argument (Default only)
/*     10/93 Changed name of AML to calplot (Calcomp plotter)
/*          and added numerous options.
/*     11/93 Added opaque option
/*      4/94 Calplot now works with map compositions.  (D. Gilliam)
/*      5/95 Major changes for RasterGraphics electrostatic plotter
/*
/*********************************************************
/*
/* LOCAL VARIABLES:
/*
/*    PATH :     System Path to /tmp filesystem
/*    NUMCOPIES : Set default to one copy of plot
/*    EXTEN :    Filename extension for Calcomp plot file
/*               (not required)
/*    OUTPUT :   Output filename
/*
/*********************************************************
/*
&ARGS PLTFIL FORMAT BANNERFLAG NUMCOPIES SCALE OPAQUE
&setvar path = /tmp/
&setvar OUTPUT = %PATH%%PLTFIL%.plt
/*
/* Check for input errors and assorted other mish-mash
/*
&if [null %PLTFIL%] &then &goto usage
&if [null %FORMAT%] &then &setvar FORMAT = 'HPGL'
&if [keyword %FORMAT% HPGL hpgl HPGL-2 hpgl-2 CALCOMP calcomp]  < = 0 &then
```

```
  &goto usage
&if ^ [exists %PLTFIL%.gra -file] and ^ [exists %PLTFIL%.plt -file] ~
  and ^ [exists %PLTFIL% -file] and ^ [exists %PLTFIL% -directory] &then ~
  &return &inform Plot file or composition %PLTFIL% does not exist.
&if [null %BANNERFLAG%] &then &setvar BANNERFLAG = 'BANNER'
&if [keyword %BANNERFLAG% BANNER banner NOBANNER nobanner] < = 0 &then
  &goto usage
&if [null %NUMCOPIES%] &then &setvar NUMCOPIES := 1
&if %NUMCOPIES% gt 5 &then
  &do
  &type You have requested %NUMCOPIES% copies.
  &sv question = [query 'Did you REALLY want this many copies' .TRUE.]
  &if %question% eq .FALSE. &then &return &inform Exiting...
  &end
&if %NUMCOPIES% le 0 &then &setvar NUMCOPIES := 1
&if [null %SCALE%] &then &setvar scale = 1.0
&if %SCALE% le 0 &then &return ~
  &inform Scale value must be greater than 0.
&if [null %OPAQUE%] &then &setvar opaque = 'NOOPAQUE'
&if [keyword %OPAQUE% NOOPAQUE OPAQUE] < = 0 &then ~
  &return &inform Invalid argument for OPAQUE.
/*
/* Convert .gra or .plt file to HPGL-2 or Calcomp format
/*
&if [keyword %FORMAT% CALCOMP calcomp] < > 0 &then ~
  &do
  colorhcbs %PLTFIL% %OUTPUT% 68436 %SCALE% %OPAQUE% %BANNERFLAG% # 1 # # # $ARCHOME/plot-
ters/calibrate.dat
  &sys lp -dcs0 %OUTPUT%
  &sys echo rm %OUTPUT% > remove.files
  &sys chmod +x remove.files
  &sys at -c -f remove.files now +10 minutes
  &sys rm remove.files
  &end
&if [keyword %FORMAT% HPGL hpgl HPGL-2 hpgl-2] < > 0 &then ~
  &do
  hpgl2 %PLTFIL% %OUTPUT% # %SCALE% %OPAQUE% %BANNERFLAG% # %NUMCOPIES% $ARCHOME/plot-
ters/hpgl2.clr
  &sys lp -dcs1 %OUTPUT%
  &sys chmod +x remove.files
  &sys echo rm %OUTPUT% > remove.files
  &sys at -c -f remove.files now +10 minutes
  &sys rm remove.files
  &end
&RETURN
&END
/*
/* Display usage
/*
&label usage
&type Usage: CALPLOT  <in_graphics_file> {HPGL-2 | CALCOMP} {BANNER | NOBANNER} {number of copies}
&type              {scale} {NOOPAQUE | OPAQUE}
&return
```

The following listings were used to generate the three satellite coverage maps.

F1_73.AML Listing

```
/* F1_73.AML
/*
/* Generates All Flights aml
/*    for 3 Satellite Coverage
/*    of Full Size Maps
/*
&setvar pprog = fltpp3
&setvar mscale = 2400
&run f453b1.aml %pprog% %mscale%
cp plotbhad.gra f453b1.gra
&run f453eb1.aml %pprog% %mscale%
cp plotbhad.gra f453eb1.gra
&run f124673b2bd.aml %pprog% %mscale%
cp plotbhad.gra f124673b2bd.gra
&run f20b2.aml %pprog% %mscale%
cp plotbhad.gra f20b2.gra
&run f12473wb2.aml %pprog% %mscale%
cp plotbhad.gra f12473wb2.gra
&run f20wb2.aml %pprog% %mscale%
cp plotbhad.gra f20wb2.gra
&run f343b3.aml %pprog% %mscale%
cp plotbhad.gra f343b3.gra
&run f343sb3.aml %pprog% %mscale%
cp plotbhad.gra f343sb3.gra
&run f33c.aml %pprog% %mscale%
cp plotbhad.gra f33c.gra
&run f33t.aml %pprog% %mscale%
cp plotbhad.gra f33t.gra
&run f63bd.aml %pprog% %mscale%
cp plotbhad.gra f63bd.gra
&run f63bf.aml %pprog% %mscale%
cp plotbhad.gra f63bf.gra
&run f73bg.aml %pprog% %mscale%
cp plotbhad.gra f73bg.gra
&run f63dp.aml %pprog% %mscale%
cp plotbhad.gra f63dp.gra
&run f63b8.aml %pprog% %mscale%
cp plotbhad.gra f63b8.gra
```

F1_7S3.AML Listing

```
/* F1_7S3.AML
/*
/* Generates All Flights aml
/*    for 3 Satellite Coverage
/*    of Working Size Maps
/*
&setvar pprog = fltpps3
&setvar mscale = 9600
&run f453b1.aml %pprog% %mscale%
cp plotbhad.gra f453b1s.gra
&run f453eb1.aml %pprog% %mscale%
cp plotbhad.gra f453eb1s.gra
&run f124673b2bd.aml %pprog% %mscale%
cp plotbhad.gra f124673b2bds.gra
&run f20b2.aml %pprog% %mscale%
cp plotbhad.gra f20b2s.gra
&run f12473wb2.aml %pprog% %mscale%
cp plotbhad.gra f12473wb2s.gra
&run f20wb2.aml %pprog% %mscale%
```

```
cp plotbhad.gra f20wb2s.gra
&run f343b3.aml %pprog% %mscale%
cp plotbhad.gra f343b3s.gra
&run f343sb3.aml %pprog% %mscale%
cp plotbhad.gra f343sb3s.gra
&run f33c.aml %pprog% %mscale%
cp plotbhad.gra f33cs.gra
&run f33t.aml %pprog% %mscale%
cp plotbhad.gra f33ts.gra
&run f63bd.aml %pprog% %mscale%
cp plotbhad.gra f63bds.gra
&run f63bf.aml %pprog% %mscale%
cp plotbhad.gra f63bfs.gra
&run f73bg.aml %pprog% %mscale%
cp plotbhad.gra f73bgs.gra
&run f63dp.aml %pprog% %mscale%
cp plotbhad.gra f63dps.gra
&run f63b8.aml %pprog% %mscale%
cp plotbhad.gra f63b8s.gra
```

F453B1.AML Listing

```
&args pprog mscale
cp f453b1.txt fltpp.txt
&run %pprog% %mscale% 1042600.0 318000.0 1049400.0 324000.0 none flt513+flt523+flt423+flt433
```

F453B1.TXT Listing

```
Burning Ground 1
3 Satellites Available
Test Flight 4 &
Test Flight 5
```

F453EB1.AML Listing

```
&args pprog mscale
cp f453eb1.txt fltpp.txt
&run %pprog% %mscale% 1049400.0 318000.0 1056200.0 324000.0 none flt513+flt523+flt423+flt433
```

F453EB1.TXT Listing

```
East of Burning Ground 1
3 Satellites Available
Test Flight 4 &
Test Flight 5
```

F124673B2BD.AML Listing

```
&args pprog mscale
cp f124673b2bd.txt fltpp.txt
&run %pprog% %mscale% 1036600.0 312000.0 1042600.0 318000.0 none
flt123+flt21a3+flt21g3+flt22g3+flt23g3+flt24g3+flt413+flt613+flt743
```

F124673B2BD.TXT Listing

```
Burning Ground 2 & Block D
3 Satellites Available
Test Flight 1, Test Flight 2,
Test Flight 4, Test Flight 6, &
Test Flight 7
```

F20B2.AML Listing

```
&args pprog mscale
cp f20b2.txt fltpp.txt
&run %pprog% %mscale% 1036600.0 312000.0 1042600.0 318000.0 none flt23g0+flt24g0
```

F20B2.TXT Listing

```
Burning Ground 2
Uncorrectable GPS Coverage
Test Flight 2   ONLY
```

F12473WB2.AML Listing

```
&args pprog mscale
cp f12473wb2.txt fltpp.txt
&run %pprog% %mscale% 1042600.0 312000.0 1048600.0 318000.0 none
flt123+flt21a3+flt21g3+flt22g3+flt23g3+flt24g3+flt413+flt743
```

F12473WB2.TXT Listing

```
West Burning Ground 2
3 Satellites Available
Test Flight 1, Test Flight 2,
Test Flight 4, &
Test Flight 7
```

F20WB2.AML Listing

```
&args pprog mscale
cp f20wb2.txt fltpp.txt
&run %pprog% %mscale% 1042600.0 312000.0 1048600.0 318000.0 none flt23g0+flt24g0
```

F20WB2.TXT Listing

```
West Burning Ground 2
Uncorrectable GPS Coverage
Test Flight 2   ONLY
```

F343B3.AML Listing

```
&args pprog mscale
cp f343b3.txt fltpp.txt
&run %pprog% %mscale% 1048000.0 333600.0 1054000.0 339600.0 none flt313+flt323+flt333+flt413+flt423+flt433
```

F343B3.TXT Listing

```
Burning Ground 3
3 Satellites Available
Test Flight 3 &
Test Flight 4
```

F343SB3.AML Listing

```
&args pprog mscale
cp f343sb3.txt fltpp.txt
&run %pprog% %mscale% 1048000.0 327600.0 1054000.0 333600.0 none flt313+flt323+flt333+flt413+flt423+flt433
```

F343SB3.TXT Listing

```
South Burning Ground 3
3 Satellites Available
Test Flight 3 &
Test Flight 4
```

F33C.AML Listing

```
&args pprog mscale
cp f33c.txt fltpp.txt
&run %pprog% %mscale% 1029600.0 336000.0 1035600.0 342000.0 none flt313+flt323+flt333
```

F33C.TXT Listing

```
Chemical Site
3 Satellites Available
Test Flight 3
```

F33T.AML Listing

```
&args pprog mscale
cp f33t.txt fltpp.txt
&run %pprog% %mscale% 1036000.0 337000.0 1042000.0 343000.0 none flt313+flt323+flt333
```

F33T.TXT Listing

```
Target Range
3 Satellites Available
Test Flight 3
```

F63BD.AML Listing

```
&args pprog mscale
cp f63bd.txt fltpp.txt
&run %pprog% %mscale% 1036600.0 318000.0 1042600.0 324000.0 none flt613
```

F63BD.TXT Listing

```
Block D
3 Satellites Available
Test Flight 6
```

F63BF.AML Listing

```
&args pprog mscale
cp f63bf.txt fltpp.txt
&run %pprog% %mscale% 1038800.0 324000.0 1044800.0 329600.0 none flt6223+flt633+flt643
```

F63BF.TXT Listing

```
Block F
3 Satellites Available
Test Flight 6
```

F73BG.AML Listing

```
&args pprog mscale
cp f73bg.txt fltpp.txt
&run %pprog% %mscale% 1034800.0 329600.0 1040800.0 335600.0 none flt713+flt723
```

F73BG.TXT Listing

```
Block G
3 Satellites Available
Test Flight 7
```

F63DP.AML Listing

```
&args pprog mscale
cp f63dp.txt fltpp.txt
&run %pprog% %mscale% 1052600.0 326200.0 1058600.0 332200.0 none flt6243
```

F63DP.TXT Listing

```
Disassembly Plant Area
3 Satellites Available
Test Flight 6
```

F63B8.AML Listing

```
&args pprog mscale
cp f63b8.txt fltpp.txt
&run %pprog% %mscale% 1058600.0 326200.0 1064600.0 332200.0 none flt613+flt6223+flt6243
```

F63B8.TXT Listing

```
Building 8000 Area
3 Satellites Available
Test Flight 6
```

FLTPP3.AML Listing

```
/* FLTPP3.AML
/*
/* Flight Plot Processing aml
/*    for 3 Satellite Coverage
/*    of Full Size Maps
/*
&args mscale minx miny maxx maxy gandfs:REST
&if [null %gandfs%] &then &goto usage

/* start arcplot if args are OK
arcplot

&setvar munlen = %mscale% / 12
&setvar mapbor = 0.5
&setvar mmaxx = [calc ( ( %maxx% - %minx% ) / %munlen% ) + %mapbor%]
&setvar mmaxy = [calc ( ( %maxy% - %miny% ) / %munlen% ) + %mapbor%]
&setvar linesyms = '41+49'
&setvar gpss = [before %gandfs% ' ']
&setvar flts = [after %gandfs% ' ']
&type %mscale% %munlen% %mapbor% %mmaxx% %mmaxy%
&type %gpss% %linesyms% %flts%

display 1040
plotbhad
mape %minx% %miny% %maxx% %maxy%
mapunits feet
mapscale %mscale%
maplimits 0.5 0.5 %mmaxx% %mmaxy%

/* Plot the Base Map Arcs
resel cc314 arcs cc314-id = 100
arcs cc314

/* Plot the 200 ft. by 200 ft. grid (fish net)
arcs bhadfn

/* Plot GPS Data if given
markerset plotter
```

```
lineset plotter
shadeset plotter
&if %gpss% = 'none' &then &goto flights
&do &while not [nul %gpss%]
  &setvar gps = [before %gpss% +]
  &setvar gpss = [after %gpss% +]
  &setvar linesym = [before %linesyms% +]
  &setvar linesyms = [after %linesyms% +]
  clearsel
  linesymbol %linesym%
  resel %gps% arcs %gps%-id = 1
  arcs %gps%
&end

/* Plot Flight Data if given
&label flights
&if %flts% = 'none' &then &goto legend
&do &while not [nul %flts%]
  &setvar flt = [before %flts% +]
  &setvar flts = [after %flts% +]
  &if [exists %flt%pts -cover] &then
  &do
    clearsel
    markersymbol 1
    resel %flt%pts points target = 1
    points %flt%pts
  &end

/* Plot Target Polys
  &if [exists %flt%ply -cover] &then
  &do
    clearsel
    resel %flt%ply polys target = 2
    polygonshades %flt%ply 5
    clearsel
    resel %flt%ply polys target = 3
    polygonshades %flt%ply 65
    clearsel
    resel %flt%ply polys target = 4
    polygonshades %flt%ply 9
    clearsel
    resel %flt%ply polys target = 5
    polygonshades %flt%ply 81
    clearsel
    resel %flt%ply polys target = 6
    polygonshades %flt%ply 61
    linesymbol 1
    clearsel
    arcs %flt%ply
  &end
&end

/* Print Map Title
&label legend
textquality proportional
textfont 94023
textsize .2 .175
move [calc %mmaxx% + 1.0] [calc %mmaxy% - 2.0]
textfile fltpp.txt

/* Print Release Date
move [calc %mmaxx% + 3.5] [calc %mmaxy% - 2.0]
textfile reldate.txt

/* Print Legend
&setvar legoffset = 9.2
```

```
move [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
text 'Legend'
&setvar legoffset = [calc %legoffset% + 0.3]
linesymbol 1
keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
keyseparation .15 .15
keymarker fltpppts.leg nobox
&setvar legoffset = [calc %legoffset% + 0.5]
&if %flts% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
  keyshade fltppply.leg
  &setvar legoffset = [calc %legoffset% + 3.5]
&end
&if %gpss% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
  keyline fltppline.leg
&end

/* Print Scale Bar, North Arrow, WES Logo, and Castle
linesymbol 1
&run scalebar [calc %mmaxx% + 3.0] [calc %mmaxy% - 7.5] FEET %munlen% 2 3
northarrow [calc %mmaxx% + 2.0] [calc %mmaxy% - 11.5] 1.5
&run bweslogo [calc %mmaxx% + 3.5] [calc %mmaxy% - 4.5] 1.5
&run bcastle [calc %mmaxx% + 1.0] [calc %mmaxy% - 4.25] 1.5

/* draw borders
box 0 0 [calc %mmaxx% + 6.5] [calc %mmaxy% + 0.5]
box %mapbor% %mapbor% %mmaxx% %mmaxy%

/* write out map corners
/* lower left
move 0.5 0.4
text %minx%
move 0.01 0.6
text %miny%
/* lower right
move [calc %mmaxx% - 0.4] 0.4
text %maxx%
move [calc %mmaxx% + 0.1] 0.6
text %miny%
/* upper left
move 0.5 [calc %mmaxy% + 0.1]
text %minx%
move 0.01 [calc %mmaxy% - 0.1]
text %maxy%
/* upper right
move [calc %mmaxx% - 0.4] [calc %mmaxy% + 0.1]
text %maxx%
move [calc %mmaxx% + 0.1] [calc %mmaxy% - 0.1]
text %maxy%

/* quit arcplot
quit
&return
&end

/*
/*  fltpp3 usage
/*
&label usage
  &type Usage: FLTPP3 <mapscale> <miny> <maxx> <maxy>
  &type              <gpscover1>...{+gpscovern}<fltcover1>...{+fltcovern}
  &return
```

FLTPPS3.AML Listing

```
/* FLTPPS3.AML
/*
/* Flight Plot Processing Small format aml
/*    for 3 Satellite Coverage
/*    of Working Size Maps
/*
&args mscale minx miny maxx maxy gandfs:REST
&if [null %gandfs%] &then &goto usage

/* start arcplot if args are OK
arcplot

&setvar munlen = %mscale% / 12
&setvar mapbor = 0.5
&setvar mmaxx = [calc ( ( %maxx% - %minx% ) / %munlen% ) + %mapbor%]
&setvar mmaxy = [calc ( ( %maxy% - %miny% ) / %munlen% ) + %mapbor%]
&setvar linesyms = '41+49'
&setvar gpss = [before %gandfs% ' ']
&setvar flts = [after %gandfs% ' ']
&type %mscale% %munlen% %mapbor% %mmaxx% %mmaxy%
&type %gpss% %linesyms% %flts%

display 1040
plotbhad
mape %minx% %miny% %maxx% %maxy%
mapunits feet
mapscale %mscale%
maplimits 0.5 0.5 %mmaxx% %mmaxy%

/* Plot the Base Map Arcs
resel cc314 arcs cc314-id = 100
arcs cc314

/* Plot the 200 ft. by 200 ft. grid (fish net)
arcs bhadfn

/* Plot GPS Data if given
markerset plotter
lineset plotter
shadeset plotter
&if %gpss% = 'none' &then &goto flights
&do &while not [nul %gpss%]
  &setvar gps = [before %gpss% +]
  &setvar gpss = [after %gpss% +]
  &setvar linesym = [before %linesyms% +]
  &setvar linesyms = [after %linesyms% +]
  clearsel
  linesymbol %linesym%
  resel %gps% arcs %gps%-id = 1
  arcs %gps%
&end

/* Plot Flight Data if given
&label flights
&if %flts% = 'none' &then &goto legend
&do &while not [nul %flts%]
  &setvar flt = [before %flts% +]
  &setvar flts = [after %flts% +]
  &if [exists %flt%pts -cover] &then
  &do
    clearsel
    markersymbol 1
    resel %flt%pts points target = 1
    points %flt%pts
```

```
  &end

/*  Plot Target Polys
  &if [exists %flt%ply -cover] &then
  &do
    clearsel
    resel %flt%ply polys target = 2
    polygonshades %flt%ply 5
    clearsel
    resel %flt%ply polys target = 3
    polygonshades %flt%ply 65
    clearsel
    resel %flt%ply polys target = 4
    polygonshades %flt%ply 9
    clearsel
    resel %flt%ply polys target = 5
    polygonshades %flt%ply 81
    clearsel
    resel %flt%ply polys target = 6
    polygonshades %flt%ply 61
    linesymbol 1
    clearsel
    arcs %flt%ply
  &end
&end

/* Print Map Title
&label legend
textquality proportional
textfont 94023
textsize .2 .175
move [calc %mmaxx% + 1.0] [calc %mmaxy% - 0.0]
textfile fltpp.txt

/* Print Release Date
move [calc %mmaxx% + 3.5] [calc %mmaxy% - 0.0]
textfile reldate.txt

/* Print Legend
&setvar legoffset = 3.2
move [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
text 'Legend'
&setvar legoffset = [calc %legoffset% + 0.3]
linesymbol 1
keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
keyseparation .15 .15
keymarker fltpppts.leg nobox
&setvar legoffset = [calc %legoffset% + 0.5]
&if %flts% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
  keyshade fltppply.leg
  &setvar legoffset = [calc %legoffset% + 3.5]
&end
&if %gpss% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
  keyline fltppline.leg
&end

/* Print Scale Bar, North Arrow, WES Logo, and Castle
linesymbol 1
&run scalebar [calc %mmaxx% + 3.0] [calc %mmaxy% - 2.5] FEET %munlen% 2 3
northarrow [calc %mmaxx% + 2.0] [calc %mmaxy% - 5.5] 1.5
&run bweslogo [calc %mmaxx% + 3.5] [calc %mmaxy% - 2.0] 1.5
&run bcastle [calc %mmaxx% + 1.0] [calc %mmaxy% - 1.75] 1.5
```

```
/* draw borders
box 0 0 [calc %mmaxx% + 6.5] [calc %mmaxy% + 0.5]
box %mapbor% %mapbor% %mmaxx% %mmaxy%

/* write out map corners
/* lower left
move 0.5 0.4
text %minx%
move 0.01 0.6
text %miny%
/* lower right
move [calc %mmaxx% - 0.4] 0.4
text %maxx%
move [calc %mmaxx% + 0.1] 0.6
text %miny%
/* upper left
move 0.5 [calc %mmaxy% + 0.1]
text %minx%
move 0.01 [calc %mmaxy% - 0.1]
text %maxy%
/* upper right
move [calc %mmaxx% - 0.4] [calc %mmaxy% + 0.1]
text %maxx%
move [calc %mmaxx% + 0.1] [calc %mmaxy% - 0.1]
text %maxy%

/* quit arcplot
quit
&return
&end

/*
/*  fltpps3 usage
/*
&label usage
  &type Usage: FLTPPS3 <mapscale> <minx> <miny> <maxx> <maxy>
  &type              <gpscover1>...{+gpscovern} <fltcover1>...{+fltcovern}
  &return
```

RELDATE.TXT Listing

```
Release Date:
03 Dec 1993
```

P1_73.AML Listing

```
rasplot f453b1.gra
rasplot f453eb1.gra
rasplot f124673b2bd.gra
rasplot f20b2.gra
rasplot f12473wb2.gra
rasplot f20wb2.gra
rasplot f343b3.gra
rasplot f343sb3.gra
rasplot f33c.gra
rasplot f33t.gra
rasplot f63bd.gra
rasplot f63bf.gra
rasplot f73bg.gra
rasplot f63dp.gra
rasplot f63b8.gra
```

P1_7S3.AML Listing

```
rasplot f453b1s.gra
rasplot f453eb1s.gra
rasplot f124673b2bds.gra
rasplot f20b2s.gra
rasplot f12473wb2s.gra
rasplot f20wb2s.gra
rasplot f343b3s.gra
rasplot f343sb3s.gra
rasplot f33cs.gra
rasplot f33ts.gra
rasplot f63bds.gra
rasplot f63bfs.gra
rasplot f73bgs.gra
rasplot f63dps.gra
rasplot f63b8s.gra
```

The following listings were used to generate the UXO flight data coverages.

FLT1DP.AML Listing

```
&run fltdp flt124
&run fltdp flt123
```

FLT2DP.AML Listing

```
&run fltdp flt21a4
&run fltdp flt21a3
&run fltdp flt21g4
&run fltdp flt21g3
&run fltdp flt22g3
&run fltdp flt23g4
&run fltdp flt23g3
&run fltdp flt23g0
&run fltdp flt24g4
&run fltdp flt24g3
&run fltdp flt24g0
```

FLT3DP.AML Listing

```
&run fltdp flt314
&run fltdp flt313
&run fltdp flt324
&run fltdp flt323
&run fltdp flt334
&run fltdp flt333
```

FLT4DP.AML Listing

```
&run fltdp flt414
&run fltdp flt424
&run fltdp flt423
&run fltdp flt434
&run fltdp flt433
```

FLT5DP.AML Listing

```
&run fltdp flt514
&run fltdp flt513
&run fltdp flt524
&run fltdp flt523
```

FLT6DP.AML Listing

```
&run fltdp flt614
&run fltdp flt613
&run fltdp flt6224
&run fltdp flt6223
&run fltdp flt6244
&run fltdp flt6243
&run fltdp flt634
&run fltdp flt633
&run fltdp flt644
&run fltdp flt643
```

FLT7DP.AML Listing

```
&run fltdp flt714
&run fltdp flt713
&run fltdp flt724
```

```
&run fltdp flt723
&run fltdp flt734
&run fltdp flt733
&run fltdp flt744
&run fltdp flt743
```

FLTDP.AML Listing

```
/* FLTDP.AML
/*
/* UXO Flight Data Processing aml
/*
&args flt
&if [null %flt%] &then &goto usage

&if [exists %flt%.pts -file] &then
&do
  GENERATE %flt%pts
  input %flt%.pts
  point
  quit
  BUILD %flt%pts POINT
  ADDITEM %flt%pts.pat %flt%pts.pat target 10 10 i
  ARCEDIT
  me %flt%pts
  ec %flt%pts
  ef label
  sel all
  cal target = $id
  save
  quit
&end

&if [exists %flt%.ply -file] &then
&do
  GENERATE %flt%ply
  input %flt%.ply
  polygons
  quit
  CLEAN %flt%ply
  BUILD %flt%ply POLY
  ADDITEM %flt%ply.pat %flt%ply.pat target 10 10 i
  ARCEDIT
  me %flt%ply
  ec %flt%ply
  ef label
  sel all
  cal target = $id
  save
  quit
&end
&return
&end

/*
/* fltdp usage
/*
&label usage
  &type Usage: FLTDP <fltcover>
  &return
```

FLTALLDDP.AML Listing

```
/* FLTALLDDP.AML
/*
/* Dissolve all UXO Flight Data that have overlapping polys
/*
&run fltddp flt124
&run fltddp flt21a4
&run fltddp flt23g4
&run fltddp flt334
&run fltddp flt414
&run fltddp flt434
&run fltddp flt524
&run fltddp flt614
```

FLTDDP.AML Listing

```
/*
/* UXO Flight Data Dissolving Process aml
/*
&args flt
&if [null %flt%] &then &goto usage

&if [exists %flt%.ply -file] &then
&do
/* Generate, Clean, and Build coverage from UXO data
  GENERATE %flt%t1ply
  input %flt%.ply
  polygons
  quit
  CLEAN %flt%t1ply
  BUILD %flt%t1ply POLY
/* Add and Define target with poly id
  ADDITEM %flt%t1ply.pat %flt%t1ply.pat target 10 10 i
  TABLES
  select %flt%t1ply.pat
  cal target = %flt%t1ply-id
  quit
  stop
/* Create labels for undefined polys
  CREATELABELS %flt%t1ply
/* Select and Define encapsulated undefined polys
  ARCPLOT
  clearsel
  resel %flt%t1ply polys target = 0
  writesel elimsel
  quit
  ELIMINATE %flt%t1ply %flt%t2ply KEEPEDGE POLY elimsel BORDER
/* Change target for world and Dissolve edge undefined polys
  TABLES
  select %flt%t2ply.pat
  resel %flt%t2ply# = 1
  cal target = -1
  quit
  stop
  DISSOLVE %flt%t2ply %flt%t3ply #all POLY
/* Change target for world and Defined edge undefined polys
  TABLES
  select %flt%t3ply.pat
  resel %flt%t3ply# = 1
  cal target = 0
  quit
  stop
  ARCPLOT
  clearsel
```

```
  resel %flt%t3ply polys target = 0
  writesel elimsel
  quit
  ELIMINATE %flt%t3ply %flt%dply NOKEEPEDGE POLY elimsel BORDER
/* Kill temp coverages and Remove temp select file
  KILL %flt%t1ply all
  KILL %flt%t2ply all
  KILL %flt%t3ply all
  rm elimsel
&end
&return
&end

/*
/* fltddp usage
/*
&label usage
  &type Usage: FLTDDP <fltcover>
  &return
```

The following listings were used to generate the GPS flight data coverages.

GPSALLDP.AML Listing

```
/*
/* Generates GPS Flight Line Coverages
/*
&run gpsdp a0712121
&run gpsdp a0712171
&run gpsdp a0713131
&run gpsdp a0713161
&run gpsdp a0714181
&run gpsdp a0714221
&run gpsdp a0715151
&run gpsdp a07flbdr
```

GPSDP.AML Listing

```
/*
/* GPS Data Processing aml
/*
&args gps
&if [null %gps%] &then &goto usage

&if [exists %gps%.gen -file] &then
&do
  GENERATE %gps%
  input %gps%.gen
  line
  quit
  BUILD %gps% LINE
  ADDITEM %gps%.aat %gps%.aat target 10 10 i
&end
&return
&end

/*
/* fltdp usage
/*
&label usage
  &type Usage: GPSDP gpscover
  &return
```

TFALLBDP.AML Listing

```
/*
/* Generates GPS Flight Line Buffered Coverages
/*
/* Buffer Individual GPS Flight Lines
/*
&run tfbdp a0712121 a071212b 47
&run tfbdp a0712171 a071217b 47
&run tfbdp a0713131 a071313b 47
&run tfbdp a0713161 a071316b 47
&run tfbdp a0714181 a071418b 72
&run tfbdp a0714221 a071422b 72
&run tfbdp a0715151 a071515b 72
/*
/* Buffer Border
/*
&run tfbdp a07flbdr a07flbdb 47
/*
/* Buffer Combinations of GPS Flight Lines
/*
```

```
&run tf12467b
&run tf3b
&run tf34b
&run tf45b
&run tf6b
&run tf7b
```

TFBDP.AML Listing

```
/*
/* GPS Test Flight Buffer Data Processing aml
/*
&args flt tfb bufzise
&if [null %flt%] &then &goto usage

&if [exists %flt% -cover] &then
&do
  BUFFER %flt% %tfb% # # %bufzise% # LINE
  CLEAN %tfb%
  BUILD %tfb% POLY
  ADDITEM %tfb%.pat %tfb%.pat target 10 10 i
  ARCEDIT
  me %tfb%
  ec %tfb%
  ef label
  sel all
  cal target = $id
  save
  quit
&end
&return
&end

/*
/* tfbdp usage
/*
&label usage
  &type Usage: TFBDP <fltlinecover> <fltbufcover> <bufsize>
  &return
```

TF12467B.AML Listing

```
/*
/* Combining of Border with GPS Test Flight Line(s) 1, 2, 4, 6, & 7
/*   then Buffering of Combination
/*

/* Check for existence of coverage
&if [exists a07f12467l -cover] &then &goto fileexists
/* Append border and flight lines 1, 2, & 4 together, Clean, and Build
APPEND a07f1241 a07flbdr ALL
a07flbdr
a0712121
a0712171
a0713161
end
CLEAN a07f1241
BUILD a07f1241 LINE
/* Set width of buffer
TABLES
select a07f1241.aat
cal target = 47
quit
```

```
stop
/* Append flight lines 6 & 7 together, Clean, and Build
APPEND a07f67l a0714221 ALL
a0714221
a0715151
end
CLEAN a07f67l
BUILD a07f67l LINE
/* Set width of buffer
TABLES
select a07f67l.aat
cal target = 72
quit
stop
/* Append border and flight line(s) together, Clean, and Build
APPEND a07f124671 a07f1241 ALL
a07f1241
a07f67l
end
CLEAN a07f124671
BUILD a07f124671 LINE
/* Buffer border and flight lines, Clean, and Build
BUFFER a07f124671 a07f12467b target # # # LINE
CLEAN a07f12467b
BUILD a07f12467b POLY
/* Remove temporary working files
KILL a07f1241 ALL
KILL a07f67l ALL
&return
&end

/*
/* file exists
/*
&label fileexists
  &type File a07f124671 exists!
  &return
```

TF3B.AML Listing

```
/*
/* Combining of Border with Test Flight Line(s) 3
/*   then Buffering of Combination
/*

/* Check for existence of coverage
&if [exists a07f3l -cover] &then &goto fileexists
/* Append border and flight line(s) together, Clean, and Build
APPEND a07f3l a07flbdr ALL
a07flbdr
a0713131
end
CLEAN a07f3l
BUILD a07f3l LINE
/* Set width of buffer
TABLES
select a07f3l.aat
cal target = 47
quit
stop
/* Buffer border and flight lines, Clean, and Build
BUFFER a07f3l a07f3b target # # # LINE
CLEAN a07f3b
BUILD a07f3b POLY
&return
&end
```

```
/*
/* file exists
/*
&label fileexists
  &type File a07f31 exists!
  &return
```

TF34B.AML Listing

```
/*
/*  Combining of Border with Test Flight Line(s) 3 & 4
/*    then Buffering of Combination
/*

/*  Check for existence of coverage
&if [exists a07f34l -cover] &then &goto fileexists
/*  Append border and flight line(s) together, Clean, and Build
APPEND a07f34l a07flbdr ALL
a07flbdr
a0713131
a0713161
end
CLEAN a07f34l
BUILD a07f34l LINE
/*  Set width of buffer
TABLES
select a07f34l.aat
cal target = 47
quit
stop
/*  Buffer border and flight lines, Clean, and Build
BUFFER a07f34l a07f34b target # # # LINE
CLEAN a07f34b
BUILD a07f34b POLY
&return
&end

/*
/* file exists
/*
&label fileexists
  &type File a07f34l exists!
  &return
```

TF45B.AML Listing

```
/*
/*  Combining of Border with Test Flight Line(s) 4 & 5
/*    then Buffering of Combination
/*

/*  Check for existence of coverage
&if [exists a07f45l -cover] &then &goto fileexists
/*  Append border and flight line 4 together, Clean, and Build
APPEND a07f4l a07flbdr ALL
a07flbdr
a0713161
end
CLEAN a07f4l
BUILD a07f4l LINE
/*  Set width of buffer
TABLES
select a07f4l.aat
cal target = 47
quit
```

```
stop
/* Prepare flight line 5 for appending
COPY a0714181 a07f51
CLEAN a07f51
BUILD a07f51 LINE
/* Set width of buffer
TABLES
select a07f51.aat
cal target = 72
quit
stop
/* Append border and flight line(s) together, Clean, and Build
APPEND a07f451 a07f41 ALL
a07f41
a07f51
end
CLEAN a07f451
BUILD a07f451 LINE
/* Buffer border and flight lines, Clean, and Build
BUFFER a07f451 a07f45b target # # # LINE
CLEAN a07f45b
BUILD a07f45b POLY
/* Remove temporary working files
KILL a07f41 ALL
KILL a07f51 ALL
&return
&end

/*
/* file exists
/*
&label fileexists
  &type File a07f451 exists!
  &return
```

TF6B.AML Listing

```
/*
/* Combining of Border with Test Flight Line(s) 6
/*   then Buffering of Combination
/*

/* Check for existence of coverage
&if [exists a07f61 -cover] &then &goto fileexists
/* Append border and flight line(s) together, Clean, and Build
APPEND a07f61 a07flbdr ALL
a07flbdr
a0714221
end
CLEAN a07f61
BUILD a07f61 LINE
/* Set width of buffer
TABLES
select a07f61.aat
cal target = 72
quit
stop
/* Buffer border and flight lines, Clean, and Build
BUFFER a07f61 a07f6b target # # # LINE
CLEAN a07f6b
BUILD a07f6b POLY
&return
&end

/*
/* file exists
```

```
/*
&label fileexists
  &type File a07f6l exists!
  &return
```

TF7B.AML Listing

```
/*
/* Combining of Border with Test Flight Line(s) 7
/*   then Buffering of Combination
/*

/* Check for existence of coverage
&if [exists a07f7l -cover] &then &goto fileexists
/* Append border and flight line(s) together, Clean, and Build
APPEND a07f7l a07flbdr ALL
a07flbdr
a0715151
end
CLEAN a07f7l
BUILD a07f7l LINE
/* Set width of buffer
TABLES
select a07f7l.aat
cal target = 72
quit
stop
/* Buffer border and flight lines, Clean, and Build
BUFFER a07f7l a07f7b target # # # LINE
CLEAN a07f7b
BUILD a07f7b POLY
&return
&end

/*
/* file exists
/*
&label fileexists
  &type File a07f7l exists!
  &return
```

The following listings were used to generate the Digital Images' Locations and Labels.

M4DEFINE.AML Listing

```
tables
define m4alllab
label
10
10
c
m4allply-id
4
5
b

/* q stop
```

M4TAB.AML Listing

```
tables
sel m4alllab
add from m4all.lab
q stop
```

M4ALL.LAB Listing

```
M0101 101
M0102 102
M0103 103
M0104 104
M0105 105
M0106 106
M0107 107
M0108 108
M0109 109
M0110 110
M0201 201
M0202 202
M0203 203
M0204 204
M0205 205
M0206 206
M0207 207
M0208 208
M0209 209
M0210 210
M0301 301
M0302 302
M0303 303
M0304 304
M0305 305
M0306 306
M0307 307
M0308 308
M0309 309
M0310 310
M0401 401
M0402 402
M0403 403
M0404 404
M0405 405
M0406 406
M0407 407
M0408 408
```

```
M0409 409
M0410 410
M0501 501
M0502 502
M0503 503
M0504 504
M0505 505
M0506 506
M0507 507
M0508 508
M0509 509
M0510 510
M0601 601
M0602 602
M0603 603
M0604 604
M0605 605
M0606 606
M0607 607
M0608 608
M0609 609
M0610 610
M0701 701
M0702 702
M0703 703
M0704 704
M0705 705
M0706 706
M0707 707
M0708 708
M0709 709
M0710 710
M0801 801
M0802 802
M0803 803
M0804 804
M0805 805
M0806 806
M0807 807
M0808 808
M0809 809
M0810 810
M0901 901
M0902 902
M0903 903
M0904 904
M0905 905
M0906 906
M0907 907
M0908 908
M0909 909
M0910 910
M1001 1001
M1002 1002
M1003 1003
M1004 1004
M1005 1005
M1006 1006
M1007 1007
M1008 1008
M1009 1009
M1010 1010
M1011 1011
M1012 1012
M1013 1013
M1014 1014
```

```
M1015 1015
M1016 1016
M1101 1101
M1102 1102
M1103 1103
M1104 1104
M1105 1105
M1106 1106
M1107 1107
M1108 1108
M1109 1109
M1110 1110
M1201 1201
M1202 1202
M1203 1203
M1204 1204
M1205 1205
M1206 1206
M1207 1207
M1208 1208
M1209 1209
M1210 1210
M1211 1211
M1212 1212
M1213 1213
M1214 1214
M1301 1301
M1302 1302
M1303 1303
M1304 1304
M1305 1305
M1306 1306
M1307 1307
M1308 1308
M1309 1309
M1310 1310
M1311 1311
M1312 1312
M1313 1313
M1314 1314
M1315 1315
M1316 1316
```

M4DP.AML Listing

```
&run fltdp m4all
ARCEDIT
me m4allply
ec m4allply
ef label
sel all
cal target = 8
save
quit
joinitem m4allply.pat m4alllab m4allply.pat M4ALLPLY-ID target
&return
```

The following listings were used to generate the Test Flight Coverage Map.

FBCMS.AML Listing

```
&setvar pprog = fcmpps
&setvar mscale = 48000
&run fbcmbhad.aml %pprog% %mscale%
cp plotbhad.gra fbcmbhad.gra
```

FBCMBHAD.AML Listing

```
&args pprog mscale
cp fbcmbhad.txt fltpp.txt
&run %pprog% %mscale% 1028400.0 311600.0 1065800.0 343000.0 none a071212b+a071217b+a071-
313b+a071316b+a071418b+a071422b+a071515b
```

FBCMBHAD.TXT Listing

```
Test Flight
Coverage Map
```

FCMPPS.AML Listing

```
/* FCMPPS.AML
/*
/* Flight Map Coverage Plot Processing Small format aml
/*
&args mscale minx miny maxx maxy gandfs:REST
&if [null %gandfs%] &then &goto usage

/* start arcplot if args are OK
arcplot

&setvar munlen = %mscale% / 12
&setvar mapbor = 0.5
&setvar mmaxx = [calc ( ( %maxx% - %minx% ) / %munlen% ) + %mapbor%]
&setvar mmaxy = [calc ( ( %maxy% - %miny% ) / %munlen% ) + %mapbor%]
&setvar linesyms = '41+49'
&setvar gpss = [before %gandfs% ' ']
&setvar flts = [after %gandfs% ' ']
&type %mscale% %munlen% %mapbor% %mmaxx% %mmaxy%
&type %gpss% %linesyms% %flts%

display 1040
plotbhad
mape %minx% %miny% %maxx% %maxy%
mapunits feet
mapscale %mscale%
maplimits 0.5 0.5 %mmaxx% %mmaxy%

/* Plot the Base Map Arcs
resel cc314 arcs cc314-id = 100
arcs cc314

/* Plot GPS Data if given
markerset plotter
lineset plotter
shadeset plotter
&if %gpss% = 'none' &then &goto flights
&do &while not [nul %gpss%]
  &setvar gps = [before %gpss% +]
  &setvar gpss = [after %gpss% +]
  &setvar linesym = [before %linesyms% +]
  &setvar linesyms = [after %linesyms% +]
```

```
  clearsel
  linesymbol %linesym%
  resel %gps% arcs %gps%-id = 1
  arcs %gps%
&end

/* Plot Test Flight Coverage Data if given
&label flights
&if %flts% = 'none' &then &goto legend
&do &while not [nul %flts%]
  &setvar flt = [before %flts% +]
  &setvar flts = [after %flts% +]
  &if [exists %flt%pts -cover] &then
  &do
    clearsel
    markersymbol 1
    resel %flt%pts points target = 1
    points %flt%pts
  &end
  &if [exists %flt% -cover] &then
  &do
    clearsel
    resel %flt% polys target = 1
    polygonshades %flt% 85
    clearsel
    arcs %flt%
  &end
&end

/* Print Map Title
&label legend
textquality proportional
textfont 94023
textsize .2 .175
move [calc %mmaxx% + 1.0] [calc %mmaxy% - 0.0]
textfile fltpp.txt

/* Print Release Date
move [calc %mmaxx% + 3.5] [calc %mmaxy% - 0.0]
textfile reldate.txt

/* Print Legend
&setvar legoffset = 3.2
move [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
text 'Legend'
&setvar legoffset = [calc %legoffset% + 0.3]
linesymbol 1
keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
keyseparation .15 .15
&setvar legoffset = [calc %legoffset% + 0.5]
&if %flts% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
  keyshade fcmply.leg
  &setvar legoffset = [calc %legoffset% + 3.5]
&end
&if %gpss% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
  keyline fltppline.leg
&end


/* Print Scale Bar, North Arrow, WES Logo, and Castle
linesymbol 1
&run scalebar [calc %mmaxx% + 3.0] [calc %mmaxy% - 2.5] FEET %munlen% 2 3
```

```
northarrow [calc %mmaxx% + 2.0] [calc %mmaxy% - 5.5] 1.5
&run bweslogo [calc %mmaxx% + 3.5] [calc %mmaxy% - 2.0] 1.5
&run bcastle [calc %mmaxx% + 1.0] [calc %mmaxy% - 1.75] 1.5

/* draw borders
box 0 0 [calc %mmaxx% + 6.5] [calc %mmaxy% + 0.5]
box %mapbor% %mapbor% %mmaxx% %mmaxy%

/* write out map corners
/* lower left
move 0.5 0.4
text %minx%
move 0.01 0.6
text %miny%
/* lower right
move [calc %mmaxx% - 0.4] 0.4
text %maxx%
move [calc %mmaxx% + 0.1] 0.6
text %miny%
/* upper left
move 0.5 [calc %mmaxy% + 0.1]
text %minx%
move 0.01 [calc %mmaxy% - 0.1]
text %maxy%
/* upper right
move [calc %mmaxx% - 0.4] [calc %mmaxy% + 0.1]
text %maxx%
move [calc %mmaxx% + 0.1] [calc %mmaxy% - 0.1]
text %maxy%

/* quit arcplot
quit
&return
&end

/*
/*  fcmpps usage
/*
&label usage
  &type Usage: FCMPPS <mapscale> <minx> <miny> <maxx> <maxy>
  &type              <gpscover1>...{+gpscovern} <fltcover1>...{+fltcovern}
  &return
```

FCMPLY.LEG Listing

```
.85
Test Flight
Coverage
```

The following listings were used to generate the Contamination Map Coverage.

CCMSL.AML Listing

```
&setvar pprog = ccmppsl
&setvar mscale = 48000
&run ccmbhad.aml %pprog% %mscale%
cp plotbhad.gra ccmbhad.gra
```

CCMBHAD.AML Listing

```
&args pprog mscale
cp ccmbhad.txt fltpp.txt
&run %pprog% %mscale% 1028400.0 311600.0 1065800.0 343000.0 none ccm13
```

CCMBHAD.TXT Listing

```
Contamination Map
Coverage
```

CCMPPSL.AML Listing

```
/* CCMPPSL.AML
/*
/* Contamination Map Coverage Plot Processing Small format aml
/*
&args mscale minx miny maxx maxy gandfs:REST
&if [null %gandfs%] &then &goto usage

/* start arcplot if args are OK
arcplot

&setvar munlen = %mscale% / 12
&setvar mapbor = 0.5
&setvar mmaxx = [calc ( ( %maxx% - %minx% ) / %munlen% ) + %mapbor%]
&setvar mmaxy = [calc ( ( %maxy% - %miny% ) / %munlen% ) + %mapbor%]
&setvar linesyms = '41+49'
&setvar gpss = [before %gandfs% ' ']
&setvar flts = [after %gandfs% ' ']
&type %mscale% %munlen% %mapbor% %mmaxx% %mmaxy%
&type %gpss% %linesyms% %flts%

display 1040
plotbhad
mape %minx% %miny% %maxx% %maxy%
mapunits feet
mapscale %mscale%
maplimits 0.5 0.5 %mmaxx% %mmaxy%

/* Plot the Base Map Arcs
resel cc314 arcs cc314-id = 100
arcs cc314

/* Plot GPS Data if given
markerset plotter
lineset plotter
shadeset plotter
&if %gpss% = 'none' &then &goto flights
&do &while not [nul %gpss%]
  &setvar gps = [before %gpss% +]
  &setvar gpss = [after %gpss% +]
  &setvar linesym = [before %linesyms% +]
  &setvar linesyms = [after %linesyms% +]
  clearsel
```

```
  linesymbol %linesym%
  resel %gps% arcs %gps%-id = 1
  arcs %gps%
&end

/* Plot Contamination Map Coverage if given
&label flights
&if %flts% = 'none' &then &goto legend
&do &while not [nul %flts%]
  &setvar flt = [before %flts% +]
  &setvar flts = [after %flts% +]
  &if [exists %flt%pts -cover] &then
  &do
    clearsel
    markersymbol 1
    resel %flt%pts points target = 1
    points %flt%pts
  &end
  &if [exists %flt%ply -cover] &then
  &do
    linesymbol 9
    clearsel
    resel %flt%ply polys target = 20
    arcs %flt%ply
    polygonshades %flt%ply 69

/* Label Contamination Maps
    textquality proportional
    textfont 94023
    textsize .2 .175
    labeltext %flt%ply label # CC NOROTATION
    linesymbol 1
    clearsel
  &end
&end

/* Print Map Title
&label legend
textquality proportional
textfont 94023
textsize .2 .175
move [calc %mmaxx% + 1.0] [calc %mmaxy% - 0.0]
textfile fltpp.txt

/* Print Release Date
move [calc %mmaxx% + 4.5] [calc %mmaxy% - 0.0]
textfile reldate4.txt

/* Print Legend
&setvar legoffset = 3.2
move [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
text 'Legend'
&setvar legoffset = [calc %legoffset% + 0.3]
linesymbol 1
keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
keyseparation .15 .15
&setvar legoffset = [calc %legoffset% + 0.5]
&if %flts% ne 'none' &then
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
    linesymbol 9
  keyshade ccmply.leg
    linesymbol 1
  &setvar legoffset = [calc %legoffset% + 3.5]
&end
&if %gpss% ne 'none' &then
```

```
&do
  keyposition [calc %mmaxx% + 3.5] [calc %mmaxy% - %legoffset%]
  keyline fltppline.leg
&end

/* Print Scale Bar, North Arrow, WES Logo, and Castle
linesymbol 1
&run scalebar [calc %mmaxx% + 3.0] [calc %mmaxy% - 2.5] FEET %munlen% 2 3
northarrow [calc %mmaxx% + 2.0] [calc %mmaxy% - 5.5] 1.5
&run bweslogo [calc %mmaxx% + 3.5] [calc %mmaxy% - 2.0] 1.5
&run bcastle [calc %mmaxx% + 1.0] [calc %mmaxy% - 1.75] 1.5

/* draw borders
box 0 0 [calc %mmaxx% + 6.5] [calc %mmaxy% + 0.5]
box %mapbor% %mapbor% %mmaxx% %mmaxy%

/* write out map corners
/* lower left
move 0.5 0.4
text %minx%
move 0.01 0.6
text %miny%
/* lower right
move [calc %mmaxx% - 0.4] 0.4
text %maxx%
move [calc %mmaxx% + 0.1] 0.6
text %miny%
/* upper left
move 0.5 [calc %mmaxy% + 0.1]
text %minx%
move 0.01 [calc %mmaxy% - 0.1]
text %maxy%
/* upper right
move [calc %mmaxx% - 0.4] [calc %mmaxy% + 0.1]
text %maxx%
move [calc %mmaxx% + 0.1] [calc %mmaxy% - 0.1]
text %maxy%

/* quit arcplot
quit
&return
&end

/*
/*  ccmppsl usage
/*
&label usage
  &type Usage: CCMPPSL <mapscale> <minx> <miny> <maxx> <maxy>
  &type              <gpscover1>...{+gpscovern} <fltcover1>...{+fltcovern}
  &return
```

CCMPLY.LEG Listing

```
.69
Contamination
Maps
```

CCMDEFINE.AML Listing

```
tables
define ccm13lab
label
10
10
```

```
c
ccm13ply-id
4
5
b

/* q stop
```

CCMTAB.AML Listing

```
tables
sel ccm13lab
add from ccm13.lab
q stop
```

CCM13.LAB Listing

```
M01 1
M02 2
M03 3
M04 4
M05 5
M06 6
M07 7
M08 8
M09 9
M10 10
M11 11
M12 12
M13 13
```

CCMDP.AML Listing

```
&run fltdp ccm13
ARCEDIT
me ccm13ply
ec ccm13ply
ef label
sel all
cal target = 20
save
quit
joinitem ccm13ply.pat ccm13lab ccm13ply.pat CCM13PLY-ID target
&return
```

The following listings were used to generate the color Flight Line Map.

A07ALLDP.AML Listing

```
&run a07dp a071212a
&run a07dp a071217a
&run a07dp a071313a
&run a07dp a071316a
&run a07dp a071418a
&run a07dp a071422a
&run a07dp a071515a
```

A07DP.AML Listing

```
/*
/*  Flight Line Points Data Processing aml
/*
&args flt
&if [null %flt%] &then &goto usage

&if [exists %flt%.gen -file] &then
&do
  GENERATE %flt%
  input %flt%.gen
  point
  quit
  BUILD %flt% POINT
&end
&return
&end

/*
/*  a07dp usage
/*
&label usage
  &type Usage: A07DP <fltptscover>
  &return
```

CFLMPPS.AML Listing

```
/*  CFLMPPS.AML
/*
/*  Color Flight Line Map Plot Processing Small format aml
/*

/* start arcplot
arcplot

display 1040
plot4
mapunits feet
mapscale 57600
mape 1025200 307700 1076000 346000

/*  Plot the Base Map Arcs
resel cc314 arcs cc314-id = 100
arcs cc314
```

```
/* Plot Flight Lines in Color
symbolset color
clearsel
markersymbol 1
resel a071212a points a071212a-id = 1
points a071212a
clearsel
markersymbol 2
resel a071217a points a071217a-id = 1
points a071217a
clearsel
markersymbol 3
resel a071313a points a071313a-id = 1
points a071313a
clearsel
markersymbol 4
resel a071316a points a071316a-id = 1
points a071316a
clearsel
markersymbol 5
resel a071418a points a071418a-id = 1
points a071418a
clearsel
markersymbol 6
resel a071422a points a071422a-id = 1
points a071422a
clearsel
markersymbol 10
resel a071515a points a071515a-id = 1
points a071515a

/* Print Legend
textquality proportional
textfont 94023
textsize .07 .07
move 6.0 1.9
text 'Legend'
markersize 1
markerscale 4
keyposition 6.0 1.8
keybox .07 .07
keyseparation .02 .02
keymarker cflm.leg nobox

/* quit arcplot
quit
&return
&end
```

CFLM.LEG Listing

```
.1
Test Flight
Number 1  (12 JUL 93)
.2
Test Flight
Number 2  (12 JUL 93)
.3
Test Flight
Number 3  (13 JUL 93)
.4
Test Flight
Number 4  (13 JUL 93)
.5
Test Flight
Number 5  (14 JUL 93)
```

```
.6
Test Flight
Number 6  (14 JUL 93)
.10
Test Flight
Number 7  (15 JUL 93)
```

Appendix C
Integration of Satellite Information Source Code

C source code listings for the Black Hills Army Depot used to integrate the number of satellites available into the ARCINFO and GPS data files are presented in this appendix.

AGAG.C Listing

```
/*          Data Processing Program for Conversion from
              Original ARCINFO and GPS files to
              ARCINFO and GPS files with Satellite Information
                    August 16, 1993
*/
/* filename:  agag.c */
/*
          usage: agag infile1.ext infile2.ext infile3.ext
                       outfile1.ext outfile2.ext outfile3.ext
*/

#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

/* error flags */
#define ERRARG    (-1)
#define ERRFILE   (-2)
#define ERREOF  (-3)
#define ERRMG   (-4)
#define ERRGR   (-5)

/* number of arguments for agag program */
#define NUMARG    6
/* number of header lines in GPS file */
#define HDR1LINES 19

main (int argc, char *argv[])
{
          char stemp[80];
          double east1, east2, east3, north1, north2, north3;
          FILE *infile1, *infile2, *infile3, *outfile1, *outfile2, *outfile3;
          float gs, geohgt1, geohgt2, geohgt3;
          int i, lastline, uncp;
          int gh, gm;
          int check_argc(int);
          int readarc(FILE *, FILE *, double *, double *, float *);
          int **allo_imat(int, int);

          /* check args */
          if(check_argc(argc))
          {
                    puts\
          ("  usage: agag infile1.ext infile2.ext infile3.ext");
                    puts
          ("           outfile1.ext outfile2.ext outfile3.ext");
                    puts("    infile1.ext  = GEN arcinfo 4 satellite file");
                    puts("    infile2.ext  = GEN arcinfo 3 satellite file");
                    puts("    infile3.ext  = GPS file");
                    puts("    outfile1.ext = GNC corrected 4 satellite file");
                    puts("    outfile2.ext = GNC corrected 3 satellite file");
                    puts("    outfile3.ext = GPC corrected gps file");
                    exit(ERRARG);
          }

          /* check files */
          if ((infile1=fopen(argv[1],"r")) == NULL)
          {
                    printf("Cannot Open %s\n",argv[1]);
                    exit(ERRFILE);
          }

          if ((infile2=fopen(argv[2],"r")) == NULL)
```

```
{
        printf("Cannot Open %s\n",argv[2]);
        exit(ERRFILE);
}

if ((infile3=fopen(argv[3],"r")) == NULL)
{
        printf("Cannot Open %s\n",argv[3]);
        exit(ERRFILE);
}

if ((outfile1=fopen(argv[4],"w")) == NULL)
{
        printf("Cannot Open %s\n",argv[4]);
        exit(ERRFILE);
}

if ((outfile2=fopen(argv[5],"w")) == NULL)
{
        printf("Cannot Open %s\n",argv[5]);
        exit(ERRFILE);
}

if ((outfile3=fopen(argv[6],"w")) == NULL)
{
        printf("Cannot Open %s\n",argv[6]);
        exit(ERRFILE);
}


/* copy first line of GEN files to GNC files */
fgets(stemp, sizeof(stemp), infile1);
fputs(stemp, outfile1);
fgets(stemp, sizeof(stemp), infile2);
fputs(stemp, outfile2);

/* copy header lines of GPS file to GPC file */
for(i=1; i < = HDR1LINES; i++)
{
        fgets(stemp, sizeof(stemp), infile3);
        fputs(stemp, outfile3);
}

/* read first data line of each arcinfo file */
readarc(infile1,outfile1,&east1,&north1,&geohgt1);
readarc(infile2,outfile2,&east2,&north2,&geohgt2);

/* generate corrected arcinfo and gps files */
/* looping on gps file which contains all gps records */
lastline=0;
uncp=0;
while ((fscanf(infile3," %d:%d:%f, %lf, %lf, %f\n"\
                ,&gh,&gm,&gs,&east3,&north3,&geohgt3)) != EOF)
{
        if ((east1 == east3) && (north1 == north3))
        {
                /* output 4 satellite gps point */
                fprintf(outfile3\
                  ," %2d:%02d:%06.3f, %11.3f, %10.3f, %8.3f, %1d\n"\
                  ,gh,gm,gs,east3,north3,geohgt3,4);

                /* end line segment of 3 satellite arcinfo file if
                  last line was from there */
                if (lastline == 3)
                        fprintf(outfile2,"END\n    1\n");
```

```
                /* output 4 satellite arcinfo point */
                fprintf(outfile1,"%11.3f, %10.3f, %8.3f\n"\
                        ,east1,north1,geohgt1);

                /* set last line flag */
                lastline=4;

                /* read next line in this arcinfo file */
                if ((readarc(infile1,outfile1\
                  ,&east1,&north1,&geohgt1)) == EOF)
                        printf("%s at EOF\n",argv[1]);
        }
        else if ((east2 == east3) && (north2 == north3))
        {
                /* output 3 satellite gps point */
                fprintf(outfile3\
                  ," %2d:%02d:%06.3f, %11.3f, %10.3f, %8.3f, %1d\n"\
                  ,gh,gm,gs,east3,north3,geohgt3,3);

                /* end line segment of 4 satellite arcinfo file if
                  last line was from there */
                if (lastline == 4)
                        fprintf(outfile1,"END\n    1\n");

                /* output 3 satellite arcinfo point */
                fprintf(outfile2,"%11.3f, %10.3f, %8.3f\n"\
                        ,east2,north2,geohgt2);

                /* set last line flag */
                lastline=3;

                /* read next line in this arcinfo file */
                if ((readarc(infile2,outfile2\
                  ,&east2,&north2,&geohgt2)) == EOF)
                        printf("%s at EOF\n",argv[2]);
        }
        else
        {
                /* output uncorrected gps point */
                fprintf(outfile3\
                  ," %2d:%02d:%06.3f, %11.3f, %10.3f, %8.3f, %1d\n"\
                  ,gh,gm,gs,east3,north3,geohgt3,0);
                uncp++;
                printf("uncorrected point %d\n",uncp);

                /* end line segment of 4 satellite arcinfo file if
                  last line was from there */
                if (lastline == 4)
                        fprintf(outfile1,"END\n    1\n");

                /* end line segment of 3 satellite arcinfo file if
                  last line was from there */
                if (lastline == 3)
                        fprintf(outfile2,"END\n    1\n");

                /* set last line flag */
                lastline=0;
        }

}

/* put final END in output files */
fprintf(outfile1,"END\n");
fprintf(outfile2,"END\n");
```

```
        fclose(infile1);
        fclose(infile2);
        fclose(infile3);
        fclose(outfile1);
        fclose(outfile2);
        fclose(outfile3);
        return (0);
}

/* check args */
int check_argc(int argc)
{
        if (argc  NUMARG+1)
        {
                puts("  Error:  Too few arguments entered.");
                return(1);
        }

        if (argc  NUMARG+1)
        {
                puts("  Error:  Too many arguments entered.");
                return(1);
        }
        return(0);
}

/* read arcinfo GEN file */
int readarc(FILE *infile, FILE *outfile\
                , double *east, double *north, float *geohgt)
{
        char rtemp[80];

        if ((fgets(rtemp, sizeof(rtemp), infile)) != NULL)
        {
                if (strlen(rtemp) > 4)
                        sscanf(rtemp," %lf, %lf, %f ",east,north,geohgt);
                else
                {
                        *east=0.0;
                        *north=0.0;
                        fputs(rtemp,outfile);
                }
        }
        else
        {
                *east=0.0;
                *north=0.0;
                return(EOF);
        }
        return(0);
}
```

Appendix D
DAP to GPS
Conversion Source Code

C source code listing for the Black Hills Army Depot used to convert the DAP time stamps into GPS locations is presented in this appendix.

DGA.C Listing

```
/*          Data Processing Program for Conversion from DAP and GPS
                    to ARCINFO
                    August 16, 1993
*/
/* filename:  dga.c */
/*
          usage: dga infile1.ext infile2.ext outfile1.ext outfile2.ext
         outfile3.ext outfile4.ext outfile5.ext outfile6.ext
*/

#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

/* error flags */
#define ERRARG   (-1)
#define ERRFILE   (-2)
#define ERREOF  (-3)
#define ERRMG   (-4)
#define ERRGR   (-5)

/* number of arguments for dga program */
#define NUMARG    8
/* number of header lines in GPS file */
#define HDR1LINES 19
/* scans per seconds */
#define SPS     350
/* horizontal pixels */
#define HRZPIX   710
/* maximum number of GPS points */
#define MAXGPS 10000

main (int argc, char *argv[])
{
          char stemp[80];
          double deast, dnorth;
          double east1, east2, north1, north2;
          double ddh;
          double gdh1, gdh2;
          double fltang;
          double toffs, pixwidth, trgang, trgdist, trgeast, trgnorth;
          double **gpspts;
          double **allo_dmat(int, int);
          FILE *infile1, *infile2, *outpts, *outply, *outfile1, *outfile2;
          FILE *outfile3, *outfile4, *outfile5, *outfile6;
          float gs, geohgt1, geohgt2;
          int i, j, firstone;
          int toffh, toffm;
          int dh, dm, ds, hrzntl, strtline, localine, stopline, type;
          int gh, gm, sv, sv1, sv2;
          int check_argc(int);
          int **allo_imat(int, int);

          /* check args */
          if(check_argc(argc))
          {
                    puts\
          ("  usage: dga infile1.ext infile2.ext outfile1.ext outfile2.ext");
                    puts\
          ("         outfile3.ext outfile4.ext outfile5.ext outfile6.ext");
                    puts("    infile1.ext  = DAP file");
                    puts("    infile2.ext  = GPC corrected gps file");
```

```
            puts("    outfile1.ext = PTS 4 satellite arcinfo file");
            puts("    outfile2.ext = PLY 4 satellite arcinfo file");
            puts("    outfile3.ext = PTS 3 satellite arcinfo file");
            puts("    outfile4.ext = PLY 3 satellite arcinfo file");
            puts("    outfile5.ext = PTS uncorrected arcinfo file");
            puts("    outfile6.ext = PLY uncorrected arcinfo file");
            exit(ERRARG);
}

/* check files */
if ((infile1=fopen(argv[1],"r")) == NULL)
{
            printf("Cannot Open %s\n",argv[1]);
            exit(ERRFILE);
}

if ((infile2=fopen(argv[2],"r")) == NULL)
{
            printf("Cannot Open %s\n",argv[2]);
            exit(ERRFILE);
}

if ((outfile1=fopen(argv[3],"w")) == NULL)
{
            printf("Cannot Open %s\n",argv[3]);
            exit(ERRFILE);
}

if ((outfile2=fopen(argv[4],"w")) == NULL)
{
            printf("Cannot Open %s\n",argv[4]);
            exit(ERRFILE);
}

if ((outfile3=fopen(argv[5],"w")) == NULL)
{
            printf("Cannot Open %s\n",argv[5]);
            exit(ERRFILE);
}

if ((outfile4=fopen(argv[6],"w")) == NULL)
{
            printf("Cannot Open %s\n",argv[6]);
            exit(ERRFILE);
}

if ((outfile5=fopen(argv[7],"w")) == NULL)
{
            printf("Cannot Open %s\n",argv[7]);
            exit(ERRFILE);
}

if ((outfile6=fopen(argv[8],"w")) == NULL)
{
            printf("Cannot Open %s\n",argv[8]);
            exit(ERRFILE);
}


/* skip over header of DAP file */
fgets(stemp, sizeof(stemp), infile1);

/* read time offset(given in seconds) from DAP file */
fscanf(infile1," %d %d %lf %lf\n",&toffh,&toffm,&toffs,&pixwidth);

/* skip over next header of DAP file */
```

```
fgets(stemp, sizeof(stemp), infile1);

/* skip over header of GPS file */
for(i=1; i < = HDR1LINES; i++)
{
        fgets(stemp, sizeof(stemp), infile2);
}

/* allocate space for GPS table */
gpspts = allo_dmat(MAXGPS,5);

/* read in GPS file up to MAXGPS points */
printf("Reading GPS file.....\n");
i=0;
while ((fscanf(infile2," %d:%d:%f, %lf, %lf, %f, %d\n"\
                ,&gh,&gm,&gs,&east2,&north2,&geohgt2,&sv)) != EOF)
{
        /* load into GPS points array */
        /* [*][0] - time stamp
          [*][1] - easting
          [*][2] - northing
          [*][3] - geo height
          [*][4] - sv info */

        /* calculate and load time stamp of GPS in hours */
        gpspts[i][0]=(double)gh+(double)gm/(double)60.0\
                +(double)gs/(double)3600.0;

        /* load easting and northing and geo height */
        gpspts[i][1]=east2;
        gpspts[i][2]=north2;
        gpspts[i][3]=geohgt2;
        gpspts[i][4]=(double)sv;

        /* inc. counter and check for max */
        i++;
        if (i > = MAXGPS)
        {
                printf("GPS Table full with %d entries.\n",MAXGPS);
                exit(ERRMG);
        }
}

/* init. table pointer */
i=1;

/* set first one marker */
firstone=0;

/* read in DAP data */
while ((fscanf(infile1," %d %d %d %d %d %d %d %d\n"\
        ,&dh,&dm,&ds,&hrzntl,&strtline,&localine,&stopline\
        ,&type)) != EOF)
{
        /* calculate time stamp of DAP in hours */
        ddh=(double)dh+(double)dm/(double)60.0\
                +(double)ds/(double)3600.0;

        /* check for 0 in strtline and stopline */
        if (strtline == 0)
        {
                strtline=stopline-350;
                printf\
                ("vert. scan at %2d %2d %2d  type %2d"\
                        ,dh,dm,ds,type);
                printf("  zero strtline corrected\n");
```

```
}
if (stopline == 0)
{
        stopline=strtline+350;
        printf\
        ("vert. scan at %2d %2d %2d  type %2d"\
                ,dh,dm,ds,type);
        printf("  zero stopline corrected\n");
}

/* correct if stopline is less than strtline */
if (stopline < = strtline)
{
        printf\
        ("vert. scan at %2d %2d %2d  type %2d"\
                ,dh,dm,ds,type);
        if (strtline > localine)
        {
                strtline=stopline-350;
                printf("  high strtline corrected\n");
        }
        if (stopline < localine)
        {
                stopline=strtline+350;
                printf("  low stopline corrected\n");
        }
}

/* add location line and time offset to DAP time */
ddh=ddh+(double)(localine-strtline)\
        /((double)(stopline-strtline)*(double)3600.0);
ddh=ddh+(double)toffh+(double)toffm/(double)60.0\
        +(double)toffs/(double)3600.0;

/* init. search flag */
j=i;

/* GPS backward search */
while (gpspts[i-1][0]>  ddh)
{
        i—;
        if ( i< = 0 )
        {
                printf("Time stamp not in GPS range.\n");
                exit(ERRGR);
        }
        printf("GPS backwards search  Record #%5d  \r",i);
}

/* issue new line if search was made */
if (i != j) printf("\n");

/* init. search flag */
j=i;

/* GPS forward search */
while (ddh > gpspts[i][0])
{
        i++;
        if (i> = MAXGPS )
        {
                printf("Time stamp not in GPS range.\n");
                exit(ERRGR);
        }
        printf("GPS forwards search   Record #%5d  \r",i);
}
```

```
/* issue new line if search was made */
if (i != j) printf("\n");

/* load variables for readability */
gdh1=gpspts[i-1][0];
gdh2=gpspts[i][0];
east1=gpspts[i-1][1];
east2=gpspts[i][1];
north1=gpspts[i-1][2];
north2=gpspts[i][2];
geohgt1=(float)gpspts[i-1][3];
geohgt2=(float)gpspts[i][3];
sv1=(int)gpspts[i-1][4];
sv2=(int)gpspts[i][4];

/* calculate east and north for DAP time stamp */
deast=east1+(east2-east1)*(ddh-gdh1)/(gdh2-gdh1);
dnorth=north1+(north2-north1)*(ddh-gdh1)/(gdh2-gdh1);

/* calculate flight line angle */
fltang=atan2(north2-north1,east2-east1);

/* add horizontal correction to DAP east and north */
if(hrzntl != HRZPIX/2)
{
        if(hrzntl  HRZPIX/2)
                trgang=fltang-M_PI_2;
        else
                trgang=fltang+M_PI_2;

        trgdist=fabs((double)hrzntl-(double)(HRZPIX/2))
                *pixwidth;
        trgeast=cos(trgang)*trgdist;
        trgnorth=sin(trgang)*trgdist;
        deast=deast+trgeast;
        dnorth=dnorth+trgnorth;
}
else
{
        printf("no horizontal correction made\n");

}

/* set outpts and outply file pointers according to sv
  at the beginning of the poly or point */
if(firstone == 0)
{
        if((sv1 == 4) && (sv2 == 4))
        {
                outpts = outfile1;
                outply = outfile2;
        }
        else if ((sv1 == 0) || (sv2 == 0))
        {
                outpts = outfile5;
                outply = outfile6;
        }
        else
        {
                outpts = outfile3;
                outply = outfile4;
        }
}

/* output feature */
```

```
switch (type)
{
  case 1:
          fprintf(outpts," %2d, %11.3f, %10.3f\n"\
                  ,type,deast,dnorth);
          break;
  case 2:
          if(firstone == 0)
          {
            firstone=1;
            fprintf(outply," %2d, 0, 0\n",type);
          }
          fprintf(outply," %11.3f, %10.3f\n"\
                  ,deast,dnorth);
          break;
  case -2:
          firstone=0;
          fprintf(outply," %11.3f, %10.3f\n"\
              ,deast,dnorth);
          fprintf(outply,"END\n");
          break;
  case 3:
          if(firstone == 0)
          {
            firstone=1;
            fprintf(outply," %2d, 0, 0\n",type);
          }
          fprintf(outply," %11.3f, %10.3f\n"\
                  ,deast,dnorth);
          break;
  case -3:
          firstone=0;
          fprintf(outply," %11.3f, %10.3f\n"\
              ,deast,dnorth);
          fprintf(outply,"END\n");
          break;
  case 4:
          if(firstone == 0)
          {
            firstone=1;
            fprintf(outply," %2d, 0, 0\n",type);
          }
          fprintf(outply," %11.3f, %10.3f\n"\
                  ,deast,dnorth);
          break;
  case -4:
          firstone=0;
          fprintf(outply," %11.3f, %10.3f\n"\
              ,deast,dnorth);
          fprintf(outply,"END\n");
          break;
  case 5:
          if(firstone == 0)
          {
            firstone=1;
            fprintf(outply," %2d, 0, 0\n",type);
          }
          fprintf(outply," %11.3f, %10.3f\n"\
                  ,deast,dnorth);
          break;
  case -5:
          firstone=0;
          fprintf(outply," %11.3f, %10.3f\n"\
              ,deast,dnorth);
          fprintf(outply,"END\n");
          break;
```

```
                case 6:
                        if(firstone == 0)
                        {
                          firstone=1;
                          fprintf(outply," %2d, 0, 0\n",type);
                        }
                        fprintf(outply," %11.3f, %10.3f\n"\
                                ,deast,dnorth);
                        break;
                case -6:
                        firstone=0;
                        fprintf(outply," %11.3f, %10.3f\n"\
                            ,deast,dnorth);
                        fprintf(outply,"END\n");
                        break;
                case 7:
                        printf("type %2d is a holiday site type\n",type);
                        break;
                case -7:
                        printf("type %2d is a holiday site type\n",type);
                        break;
                case 8:
                        if(firstone == 0)
                        {
                          firstone=1;
                          fprintf(outply," %2d, 0, 0\n",type);
                        }
                        fprintf(outply," %11.3f, %10.3f\n"\
                                ,deast,dnorth);
                        break;
                case -8:
                        firstone=0;
                        fprintf(outply," %11.3f, %10.3f\n"\
                            ,deast,dnorth);
                        fprintf(outply,"END\n");
                        break;
                default:
                        printf("type %2d not found\n",type);
                }
        }

        /* put final END in output files */
        fprintf(outfile1,"END\n");
        fprintf(outfile2,"END\n");
        fprintf(outfile3,"END\n");
        fprintf(outfile4,"END\n");
        fprintf(outfile5,"END\n");
        fprintf(outfile6,"END\n");

        /* close files */
        fclose(infile1);
        fclose(infile2);
        fclose(outfile1);
        fclose(outfile2);
        fclose(outfile3);
        fclose(outfile4);
        fclose(outfile5);
        fclose(outfile6);
        return (0);
}

/* check args */
int check_argc(int argc)
{
        if (argc < NUMARG+1)
        {
```

```
                puts("  Error:  Too few arguments entered.");
                return(1);
        }

        if (argc > NUMARG+1)
        {
                puts("  Error:  Too many arguments entered.");
                return(1);
        }
        return(0);
}

/* allocates double matrix nr by nc */
double **allo_dmat(int nr, int nc)
{
        int i;
        double **dmat;

        dmat = (double **)calloc(nr, sizeof(double *));
        for(i=0; i < nr; i++)
                dmat[i] = (double *)calloc(nc, sizeof(double));
        return dmat;
}
```

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	August 1995	Final report

4. TITLE AND SUBTITLE

Remote Sensing of Surface Unexploded Ordnance at Black Hills Army Depot, Edgemont, South Dakota

5. FUNDING NUMBERS

6. AUTHOR(S)

Hollis H. (Jay) Bennett, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

U.S. Army Engineer Waterways Experiment Station
3909 Halls Ferry Road, Vicksburg, MS 39180-6199

8. PERFORMING ORGANIZATION REPORT NUMBER

Technical Report EL-95-28

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Engineer Division, Huntsville
4820 University Square
Huntsville, AL 35816-1822

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

Available from National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

The need of multiuse technology is becoming greater with the budget reductions taking place in the Department of Defense. This report describes the use of a helicopter-mounted sensing and processing system, which was originally designed for remote minefield detection, as a tool for detecting unexploded ordnance (UXO) at the terrain surface. This detection is based on the remote identification of surface anomalies and materials that may indicate the presence of UXO contamination. The system consists of an active/passive line scanner and real-time processing/display equipment. The scanner collects three channels of optically aligned image data consisting of two active laser channels, one polarized reflectance and the other total reflectance, and a passive thermal infrared channel. The real-time processing and display system makes use of a massively parallel processor. The system also incorporates onboard recording and a Global Positioning System (GPS). GPS data will allow the location of contaminated areas to be added into a Geographical Information System so this information can be overlaid on the base maps of the facility. Maps and digital images of the detected surface anomalies and materials were generated for the sites of interest at the Black Hills Ordnance Depot.

14. SUBJECT TERMS

See reverse.

15. NUMBER OF PAGES

156

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED		

14. (Concluded).

Image processing	Parallel processing	REMIDS	UXO detection
Laser	Polarization	Remote sensing	
Multispectral imaging	Real-time processing	Thermal IR	

Destroy this report when no longer needed. Do not return it to the originator.